Supplemental Materials for: A new grand mean paleomagnetic pole for the 1.11 Ga Umkondo Large Igneous Province with implications for paleogeography and the geomagnetic field

N.L. Swanson-Hysell[1,*], T.M. Kilian[1], R.E. Hanson[2]

1 Department of Earth and Planetary Science, University of California, Berkeley, CA 94720, USA
2 School of Geology, Energy, and the Environment, Texas Christian University, Fort Worth, TX 76129, USA

* swanson-hysell@berkeley.edu

Contents

1 Introduction

This Supplementary Information is output from two Jupyter notebooks that can be viewed online at the links provided below or accessed in the Github repository associated with the paper: https://github.com/Swanson-Hysell-Group/2015_Umkondo. The two notebooks compiled here include a published data compilation notebook and a new data and associated analysis notebook. The published data compilation notebook contains the details of the compilation including the combination of previously published "sites" into single cooling unit sites: http://nbviewer.jupyter.org/github/Swanson-Hysell-Group/2015_Umkondo/blob/master/Code/Umkondo_Published_Data_Compilation.ipynb. The new data and associated analysis notebook includes analysis of the newly developed data, combination of that data with previously published results and the code associated with the calculation of the mean pole and paleosecular variation parameters: http://nbviewer.jupyter.org/github/Swanson-Hysell-Group/2015_Umkondo/blob/master/Code/Umkondo_New_Data.ipynb. Some of the tables which are truncated below, can be viewed in full when the notebooks are rendered in a web browser.

In addition to the notebooks, all of the data that are imported into them for the analysis are available within the Github repository.

2 Synthesis of prior paleomagnetic results from the Umkondo Large Igneous Province

Prior paleomagnetic results from the Umkondo Large Igneous Province were analyzed using the reasoning and code detailed below.

2.1 Approach to dealing with previously published data

There are multiple groups of data being integrated into our analysis: - Group 1. Site mean data from Gose et al. (2006) wherein the measurement level data were fully documented in the theses of Pancake (2001) and Seidel (2004). - Group 2. Site mean data published by Jones and

McElhinny (1996), McElhinny and Opdyke (1964), and McElhinny (1966) and compiled by Gose et al. (2006) wherein the sample level data are not available. There are additional similar data in the Gose et al. (2006) that we do not include for reasons stated below. Results not included in Gose et al. (2006), found in Mare et al. (2006) and Wilson et al. (1987) are also included in the analysis. - Group 3. New data from Umkondo sills of Botswana from this study.

The definition of a site in Pancake (2001), Seidel (2004) and Gose et al. (2006) was a single geographic locality within a single igneous unit. In Gose et al. (2006), these sites were given equal weight in calculating 10 regional mean directions. A Fisher mean was then calculated from these regional mean directions for the overall Umkondo mean. As discussed in the main text, we consider the best practice in the calculation of a paleomagnetic pole to be taking the Fisher mean of virtual geomagnetic poles (VGPs) whereby each VGP is an individual cooling unit that is interpretted to record a single snapshot of the geomagnetic field. Within the Gose et al. (2006) analysis, there are cases where multiple sites are from within the same sill, but where each site is still given equal weight in calculating the regional mean with the regional means then being given equal weight in calculating an overall mean.

The first step in this analysis was to redo all of the principal component analysis of the Pancake (2001) and Seidel (2004) data in order to have the Group 1 data at the specimen level. Having the data at the specimen level allows for new means to be calculated for cooling units that had been sampled at multiple "sites" by these workers. Using geological maps along with our field data, we have gone through previously studied sites and assessed, to the best of our ability, whether multiple "sites" are from the same cooling unit (the definition of site used by the Magnetics Information Consortium is a unique geological unit representing an instance in time). This approach enables us to transform the data given in Pancake (2001), Seidel (2004), and Gose et al. (2006) to VGPs.

In terms of the Group 2 data sets, we use data from these studies when they meet the following conditions: (1) number of samples is greater than 3 (n>3) and (2) they can be determined to be from a single cooling unit distinct from a cooling unit where data were obtained by Pancake (2001), Seidel (2004) or our study. Here are details related to each one of these studies that fall into Group 2:

1. Jones and McElhinny (1966)— Dolerite sills from Botswana and South Africa. Sites from this study are include when we can determine that they are distinct cooling units.

    - Sites 1 and 2 are from the Shoshong Sill (sensu stricto) which was also analyzed by Pancake (2001) and in our study. Therefore, we will not use the Jones and McElhinny (1966) data for this sill.
    - Sites 3, 4 and 5 are from the same sill. All sites have excellent statistics and very similar directions. Site 3 has the lowest α_{95} (2.0) and most samples (n=8) and is the VGP we will use for the grand mean.
    - Site 6 is from the same sill as JP22/JP23/JP24/JP25 and will therefore not be used for the new grand mean.
    - Site 7 is from a unique sill in the Botswana North area.

- Site 8 is from a sill we refer to as the Lentsweletau Sill. The same sill was sampled by Swanson-Hysell and Hanson as site PW20, without yielding coherent results. Therefore, this Jones and McElhinny sill is included in the compilation.
- Site 9 is from the Kgale Peak sill which was analyzed in Pancake (2001) and in this study. Therefore, the Jones and McElhinny (1966) data will not be used for this sill.
- Site A1 is from a Paleoproterozoic Moshaneng sill.
- Site 10 is from a unique Waterberg sill.
- Site 11 is the same sill as sampled by Seidel (2004) at the W-01 and W-02 sites and is therefore not included in our compilation.
- Site 12 is the same sill as sampled by Seidel (2004) at the VF4 and VF5 sites. These data were similar in direction (and north-seeking polarity) to the VF4 and VF5 sites of Seidel, but the data from site 12 was much more tightly grouped and are preferentially included over the Seidel results. This sill was dated to be 1108.5 ± 0.8 Ma (207Pb/206Pb baddeleyite date; Hanson et al., 2004).
- Site 13 is from an Umkondo sill cross-cutting the Premier Kimberlite located to the west of the Waterberg outcrops in the Middleburg basin. The site was collected underground and is not related to any of the intrusions sampled by Seidel (2004).

2. McDonald and Anderson (1973)— Dolerite sills from the Vredefort Dome area (unpublished data from an honors thesis). None of these data have the number of samples (n) reported and are not included in our compilation.

3. McElhinny (1966)— All of these sites from the Umkondo Lavas have n<=3 except for site 4 which is identical to site J of McElhinny and Opdyke (1964; same location and direction) as stated in the 1966 paper:

> “Site 4, in the Jersey basalts, was previously sampled by McElhinny & Opdyke (1964) as Umkondo dolerite and thought to be a very fine grained variety. At the time of sampling, geological mapping in the area was still in progress. Mapping further south the following field season by Watson (1964) established that these were in fact the northerly extension of the Zamchiya basalts. The connection between the two was traced across the MoCambique border and back into Southern Rhodesia. The Zamchiya basalts often contain amygdales, whereas the Jersey basalts are the non-amygdaloidal variety. Site 4 is therefore that reported as site J by McElhinny & Opdyke (1964)and the measurements made on seven 6 in. cores at this locality are reported here again (see Table 1).”

4. McElhinny and Opdyke (1964)— Dolerites (and basalts) from Zimbabwe. Based on analysis of their locations with reference to mapping by Tyndale-Biscoe (1958), Watson (1969) and Stocklmayer (1978), we consider sites A, B, and C to come from a single cooling unit. Sites D through G each come from separate cooling units, and sites H and I come from still another cooling unit. Given that site B has an α_{95} of 4.5 and sites A and C had α_{95} values of 9.0 and 12.5, respectively, we choose to use the mean from site B as representative of the sill’s direction (without the availibility of sample level data it is difficult to combine these

data into a single site level mean). The same selection process is used for sites H and I, where H is preffered because of its higher precision.

5. Mare et al. (2006)— Dolerite sills sampled in two regions within the Nylstroom protobasin and the Middelburg basin.

    - One sill was sampled 6 times in the Nylstroom area with no overlap with any other previously published data (SHD- sample suite). However, the site mean results are very scattered and inconsistent with very few samples used to calculate a final mean. Therefore we will not include Nylstroom (Swaershoek Fm. dolerite) data in the compilation.
    - 8 sites were sampled in the Middelburg area with four of those sites overlapping with data generated from sills in Seidel (2004). Of those four, two sites have slightly inconsistent results with very large error intervals (WRD6 and WRD7). Two others sites (WRD4 and WRD5) are also from unique cooling units and have much lower error intervals.

6. Wilson et al. (1987)— Two dolerite dikes sampled in northern Zimbabwe to the WNW of sites sampled by McElhinny and Opdyke (1964).

    - One dike (Wil_1) intrudes a very prominent NNE-trending set of faults (Popoteke), east of Harare, and yielded an Umkondo magnetization.
    - Another dike (Wil_2) named the Deweras dike, trends roughly northeast and was sampled at its southern end, yielding a north-seeking Umkondo-like magentization.

2.2 Import function modules

The libraries imported below are standard scientific python libraries with the exception of the pmag.py, pmagplotlib.py and ipmag.py function libraries. These libraries are included in the Github repository with this notebook for archival purposes.

```
In [1]: import pmag, pmagplotlib, IPmag
        import matplotlib as mpl
        import matplotlib.pyplot as plt
        from mpl_toolkits.basemap import Basemap
        import pandas as pd
        pd.set_option('display.max_columns',500)
        pd.set_option('display.max_rows',500)
        from IPython.core.display import HTML
        import numpy as np
        import scipy as sp
        import pickle
        from scipy import special
        from IPython.display import display, Image
        %matplotlib inline
```

In order to have the pandas dataframes output as LaTEX tables, we include the code in the next block.

```
In [2]: def _repr_latex_(self):  return "\centering{%s}" % self.to_latex()
        pd.DataFrame._repr_latex_ = _repr_latex_
```

Some dates from Hanson et al., (2004) need to be combined into new weighted means. The function defined below will calculate these weighted means using the standard method as articulated in McLean et al., (2011).

```
In [3]: def weighted_mean(dates,sigma):

            inverse_variance=[]
            alpha=[]
            weighted_mean_calc=[]

            for n in range(0,len(dates)):
                value = 1/sigma[n]**2
                inverse_variance.append(value)
            inverse_variance_sum = np.sum(inverse_variance)

            for n in range(0,len(dates)):
                value = (1/sigma[n]**2)/inverse_variance_sum
                alpha.append(value)

            #calculate the weights for each date
            for n in range(0,len(dates)):
                value = alpha[n]*dates[n]
                weighted_mean_calc.append(value)

            #take the sum of the weights multiplied by the dates to get the weighted
            #mean (equation 64 of McLean et al., 2011)
            weighted_mean = np.sum(weighted_mean_calc)

            #the variance of the weighted mean is simply the inverse of the sum of
            #the inverse variances of each date (equation 66 of McLean et al., 2011)
            variance = 1/inverse_variance_sum

            weighted_mean_sigma = np.sqrt(variance)

            print('The weighted mean is:')

            print weighted_mean
```

```
    print('With a 2sigma error of:')

    print 2*weighted_mean_sigma
```

2.3 Compilation from Gose et al., (2006)

The data table of site mean paleomagnetic directions from the Gose et al. (2006) compilation is imported and displayed below.

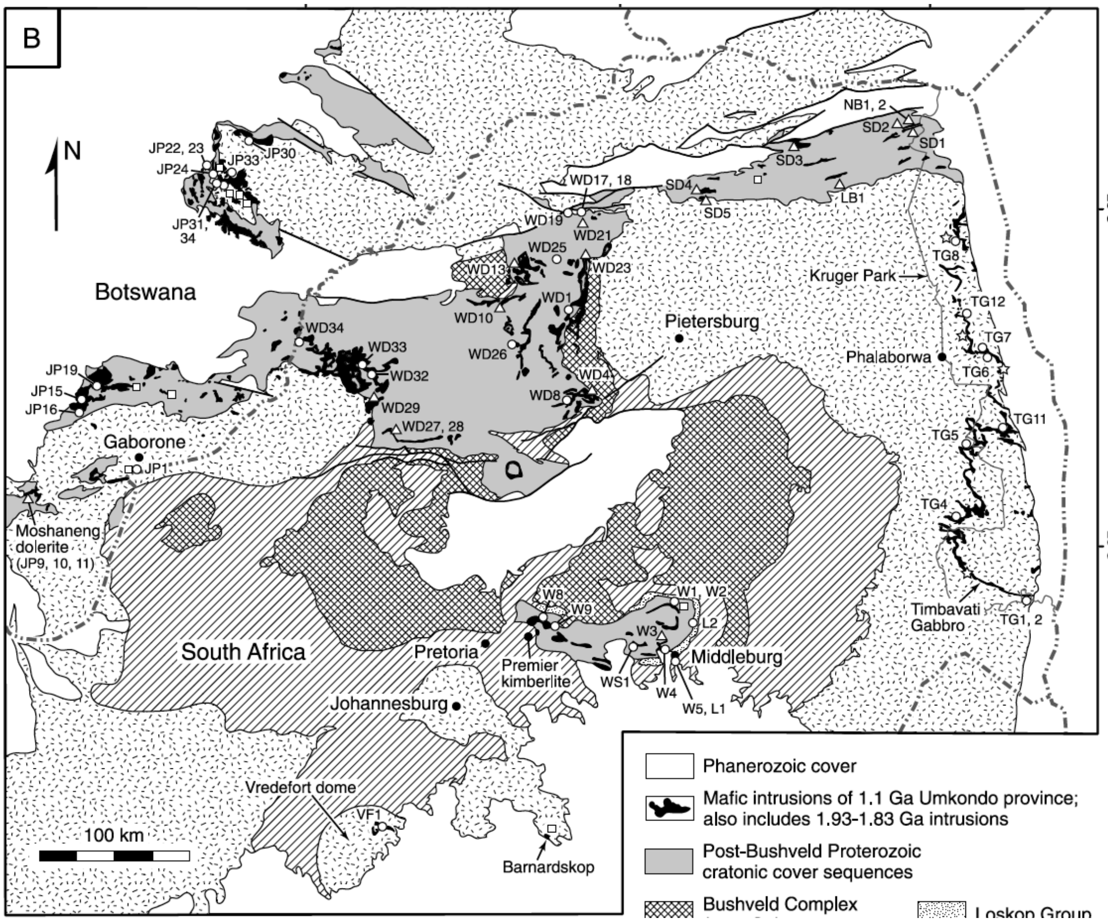
```
In [4]: Gose06_data_table=pd.read_csv('../Data/Prior_Data/Gose+06_data.csv')
        Gose06_data_table
```

Out[4]:

	Site ID	Site Lat (N)	Site Long (E)	N	R	Dec	Inc	k	a95	Lat	Long	dp	dm	Ref
0	JP1	-24.69	25.87	5/1	4.765	1.2	-0.8	17.0	19.1	65.7	28.9	9.5	19.1	Pancake [2001] ar
1	JP15	-24.32	25.53	8/2	7.914	194.4	-2.2	81.8	6.2	61.0	56.4	3.1	6.2	Pancake [2001] ar
2	JP16	-24.37	25.52	9/1	7.794	194.8	-28.2	6.6	21.6	48.0	47.1	13.0	23.7	Pancake [2001] ar
3	JP19	-24.23	25.64	5/1	4.738	196.8	-11.4	15.3	20.2	55.9	56.5	10.4	20.5	Pancake [2001] ar
4	JP9 (Moshaneng)	-24.96	25.25	14/0	13.677	324.1	7.2	40.2	6.3	45.0	329.4	3.2	6.4	Pancake [2001] ar
5	JP10 (Moshaneng)	-24.90	25.25	13/0	12.261	315.0	0.9	31.6	7.5	39.6	318.5	3.7	7.5	Pancake [2001] ar
6	JP11 (Moshaneng)	-24.94	25.30	10/10	9.871	182.3	-27.3	69.6	5.8	50.6	28.8	3.5	6.4	Pancake [2001] ar
7	J_M7	-24.33	26.13	6	5.970	193.5	-5.5	165.0	5.2	59.9	54.2	NaN	NaN	Jones and McElhi
8	J_M8	-24.23	25.87	7	6.545	191.0	-33.0	13.2	17.0	46.5	41.3	NaN	NaN	Jones and McElhi
9	J_M9	-24.67	25.88	6	5.987	187.5	-13.0	379.0	3.4	57.7	40.3	NaN	NaN	Jones and McElhi
10	A1	-24.93	25.30	4	3.988	329.5	8.5	244.0	5.9	48.2	335.5	NaN	NaN	Jones and McElhi
11	JP22	-22.92	26.38	6/1	5.676	176.9	3.3	15.4	17.6	68.6	17.9	8.8	17.6	Pancake [2001] ar
12	JP23	-22.92	26.37	4/1	3.792	178.0	-7.3	14.4	25.0	63.4	22.0	12.7	25.2	Pancake [2001] ar
13	JP24	-22.91	26.39	5/1	4.831	185.4	-10.8	23.7	16.0	61.1	37.6	8.2	16.3	Pancake [2001] ar
14	JP30	-22.70	26.61	5/1	4.898	181.3	6.9	39.3	12.3	70.7	30.6	6.2	12.4	Pancake [2001] ar
15	JP31	-23.00	26.48	8/2	7.954	190.8	-10.2	152.6	4.5	59.9	48.3	2.3	46.0	Pancake [2001] ar
16	JP33	-22.94	26.44	6/4	5.815	186.8	-19.1	27.1	13.1	56.6	38.6	7.1	13.7	Pancake [2001] ar
17	JP34	-22.98	26.46	6/1	5.894	184.7	-2.1	47.4	9.8	65.5	37.9	4.9	9.8	Pancake [2001] ar
18	J_M1	-23.10	26.60	8	7.961	190.5	14.0	180.0	4.2	70.7	62.7	NaN	NaN	Jones and McElhi
19	J_M2	-23.05	26.55	7	6.991	190.5	9.5	674.0	2.3	69.1	56.6	NaN	NaN	Jones and McElhi
20	J_M3	-23.00	26.41	8	7.991	190.5	4.0	796.0	2.0	66.6	53.7	NaN	NaN	Jones and McElhi
21	J_M4	-23.00	26.41	6	5.981	187.0	5.0	254.0	6.7	68.5	46.1	NaN	NaN	Jones and McElhi
22	J_M5	-23.00	26.41	7	6.981	187.0	5.0	323.0	3.3	38.4	45.7	NaN	NaN	Jones and McElhi
23	J_M6	-22.90	26.40	8	7.992	186.5	0.5	897.0	1.8	66.4	42.8	NaN	NaN	Jones and McElhi
24	WD1	-23.81	28.74	9	8.490	184.0	-8.5	15.7	13.4	61.7	37.3	6.8	13.5	this study Gose+
25	WD8	-24.28	28.71	12	11.489	171.4	-26.3	21.5	9.6	50.9	15.4	5.6	10.4	this study Gose+
26	WD17	-23.15	28.75	10	9.664	189.5	-18.8	26.8	9.5	55.9	45.6	5.2	9.9	this study Gose+
27	WD18	-23.15	28.75	5	4.817	190.3	-11.5	21.8	16.8	59.3	49.1	8.6	17.0	this study Gose+
28	WD19	-23.16	26.68	10	9.402	190.5	-43.5	15.1	12.9	40.4	41.2	10.0	16.0	this study Gose+
29	WD25	-23.42	28.65	8	7.013	205.6	11.9	7.1	22.4	59.9	87.4	11.5	22.7	this study Gose+
30	WD26	-23.95	28.39	13	12.456	171.7	10.6	22.1	9.0	69.8	4.0	4.6	9.1	this study Gose+
31	WD28H	-24.50	27.56	11	10.396	340.8	72.0	16.6	11.6	6.9	17.1	18.0	20.4	Hanson et al. [20
32	WD28L	-24.50	27.56	7	6.943	179.8	-38.3	104.4	5.9	44.0	27.3	4.2	7.0	Hanson et al. [20
33	WD32	-24.14	27.41	6	5.657	181.4	3.7	14.6	18.1	67.7	31.2	9.1	18.2	this study Gose+
34	WD33	-24.05	27.32	10	9.218	206.9	-36.2	11.5	14.9	38.7	60.3	10.1	17.3	this study Gose+
35	WD34	-23.84	26.93	7	6.560	158.6	-27.2	13.6	16.9	46.3	355.9	10.0	18.5	this study Gose+
36	J_M10	-22.92	29.93	5	4.940	194.0	24.0	66.5	9.5	73.1	84.2	NaN	NaN	Jones and McElhi
37	W01	-25.49	29.46	16/0	15.762	175.3	-0.5	63.1	4.7	63.9	18.8	2.3	4.7	Seidel [2004] and
38	W02	-25.47	29.44	9/9	8.265	199.1	-19.6	10.9	16.3	49.9	59.4	8.9	17.1	Seidel [2004] and
39	W03	-25.70	29.41	9/3	8.340	337.5	71.8	12.1	15.4	5.5	29.4	23.8	27.1	Seidel [2004] and
40	W04	-25.75	29.45	13/0	12.693	174.9	-10.0	39.1	6.7	58.8	19.6	3.4	6.8	Seidel [2004] and
41	W05	-25.76	29.48	11/2	10.718	174.0	-12.8	35.5	7.8	57.3	18.4	4.0	7.9	Seidel [2004] and
42	W08	-25.59	29.58	9/3	8.901	201.2	12.3	80.6	5.8	61.9	78.5	3.0	5.9	Seidel [2004] and
43	W09	-25.65	28.60	11/1	10.585	197.7	9.2	24.1	9.5	63.0	70.4	4.8	9.6	Seidel [2004] and
44	L-1	-25.76	29.48	19/0	18.414	188.0	14.5	30.7	6.2	70.1	53.4	3.2	6.3	Seidel [2004] and
45	L-2	-25.60	29.62	11/0	10.984	6.9	49.1	636.4	1.8	34.0	36.9	1.6	2.4	Seidel [2004] and
46	WS-1	-25.71	29.21	11/0	10.853	189.0	-0.5	67.9	5.6	62.6	49.2	2.8	5.6	Seidel [2004] and
47	J_M11	-25.50	29.46	6	5.633	188.0	6.0	13.6	19.0	66.2	50.2	NaN	NaN	Jones and McElhi
48	J_M12	-26.90	28.53	6	5.983	16.0	-14.5	292.0	3.9	65.3	69.3	NaN	NaN	Jones and McElhi
49	J_M13	-25.70	28.53	10	9.878	183.0	-3.0	73.5	5.7	62.7	34.6	NaN	NaN	Jones and McElhi
50	TG01	-25.35	31.81	11/13	10.933	191.1	0.9	150.1	3.7	62.9	56.8	1.9	3.7	Seidel [2004] and
51	TG01-K	-25.35	31.81	13/10	12.985	324.7	-68.1	820.5	1.4	52.8	248.9	2.0	2.4	Seidel [2004] and
52	TG02	-25.35	31.79	11/1	10.629	193.6	0.1	26.9	9.0	61.5	61.3	4.5	9.0	Seidel [2004] and
53	TG04	-24.94	31.30	10/2	9.446	188.5	7.9	16.2	12.4	67.5	53.9	6.3	12.4	Seidel [2004] and
54	TG05-AF	-24.55	31.35	8	7.448	20.5	-3.7	12.7	16.2	59.9	75.5	8.1	16.2	Seidel [2004] and
55	TG05-TD	-24.55	31.35	13/2	12.228	210.6	1.6	15.5	10.9	52.0	87.2	5.4	10.9	Seidel [2004] and
56	TG05B	-24.55	31.35	12/2	11.479	326.9	4.4	21.1	4.4	50.8	39.7	5.6	11.2	this study Gose+
57	TG06	-23.91	31.45	11/1	10.529	194.5	-1.3	21.2	10.1	61.8	63.1	5.1	10.1	Seidel [2004] and
58	TG07	-23.90	31.46	9/3	8.958	181.6	-3.5	191.6	3.7	64.3	35.2	1.9	3.7	Seidel [2004] and
59	TG08	-23.23	31.23	11/1	10.400	184.0	-20.3	16.7	11.5	56.0	38.3	6.3	12.1	Seidel [2004] and
60	TG11	-24.40	31.59	7/2	6.978	182.3	0.7	274.0	3.7	65.9	37.1	1.8	3.7	Seidel [2004] and
61	TG12	-23.72	31.31	12/1	10.928	176.9	20.8	10.3	14.2	76.7	17.9	7.9	15.0	Seidel [2004] and
62	KD01	-25.35	31.81	12/5	11.717	345.4	-67.0	38.9	7.0	63.2	233.0	9.7	11.7	Seidel [2004] and
63	Har_10	-25.20	31.20	8	NaN	190.6	-4.9	170.0	4.3	61.6	54.2	NaN	NaN	Hargraves et al. [
64	Henth	-24.00	31.30	4	NaN	185.0	1.0	70.0	11.0	66.0	43.9	NaN	NaN	Henthorn [1981]
65	VF1	-25.84	27.52	11/1	10.399	22.6	-15.3	16.6	11.5	61.9	81.5	6.1	11.8	Seidel [2004] and
66	M_A2	-27.00	27.40	NaN	NaN	14.4	-9.5	NaN	26.5	63.8	61.6	NaN	NaN	McDonald and Ar
67	M_A4	-27.10	27.60	NaN	NaN	15.4	-6.7	NaN	13.0	62.1	62.0	NaN	NaN	McDonald and Ar
68	M_A6	-26.80	27.50	NaN	NaN	353.3	-19.2	NaN	14.0	71.7	NaN	NaN	NaN	McDonald and Ar
69	M_A7	-27.00	27.40	NaN	NaN	354.2	-12.6	NaN	13.0	68.6	NaN	NaN	NaN	McDonald and Ar
70	M_A10	-27.00	27.60	NaN	NaN	344.8	-3.4	NaN	23.0	60.7	NaN	NaN	NaN	McDonald and Ar
71	M_A11	-27.10	27.60	NaN	NaN	1.7	-14.1	NaN	22.0	70.0	NaN	NaN	NaN	McDonald and Ar
72	NB2	-22.56	30.86	8/3	7.602	173.4	-53.7	17.6	13.6	32.9	24.4	13.3	19.0	this study Gose+
73	M_O_A	-18.00	32.80	9	8.757	186.0	-17.0	33.0	9.0	63.0	49.5	NaN	NaN	McElhinny and C
74	M_O_B	-18.10	32.90	5	4.985	171.5	-10.0	267.0	4.5	65.5	12.0	NaN	NaN	McElhinny and C
75	M_O_C	-18.20	32.85	8	7.671	184.5	-8.0	21.0	12.5	67.5	44.5	NaN	NaN	McElhinny and C
76	M_O_D	-18.45	32.76	10	9.288	168.0	-5.5	12.6	14.0	66.0	21.0	NaN	NaN	McElhinny and C
77	M_O_E	-19.53	32.63	10	9.902	185.0	-3.5	92.0	5.0	68.0	46.0	NaN	NaN	McElhinny and C
78	M_O_F	-19.60	32.80	8	7.966	179.5	-13.0	206.0	4.0	64.0	31.5	NaN	NaN	McElhinny and C
79	M_O_H	-19.85	32.95	10	9.576	185.0	-2.5	21.0	10.5	68.5	46.5	NaN	NaN	McElhinny and C
80	M_O_I	-19.90	32.80	8	6.755	176.0	-14.0	5.7	25.5	62.5	24.0	NaN	NaN	McElhinny and C
81	M_O_J	-20.53	32.66	7	6.570	180.5	-10.0	14.0	16.5	64.5	34.0	NaN	NaN	McElhinny and C
82	McE_1	-20.70	32.46	2	NaN	173.0	-11.0	NaN	NaN	63.0	17.0	NaN	NaN	McElhinny [1966]
83	McE_2	-20.70	32.50	1	NaN	178.0	-8.0	NaN	NaN	65.0	27.5	NaN	NaN	McElhinny [1966]
84	McE_3	-20.71	32.50	3	NaN	159.0	-19.0	NaN	NaN	53.0	356.5	NaN	NaN	McElhinny [1966]
85	McE_4	-20.53	32.66	7	NaN	180.5	-10.0	NaN	NaN	64.5	34.0	NaN	NaN	McElhinny [1966]
86	McE_5	-20.33	32.15	3	NaN	200.0	-5.0	NaN	NaN	60.0	75.0	NaN	NaN	McElhinny [1966]
87	McE_6	-20.25	32.15	1	NaN	177.0	14.0	NaN	NaN	76.5	45.0	NaN	NaN	McElhinny [1966]
88	McE_7	-20.16	32.18	1	NaN	142.0	-36.0	NaN	NaN	69.5	65.5	NaN	NaN	McElhinny [1966]
89	McE_8	-20.06	31.29	2	NaN	142.0	-36.0	NaN	NaN	35.5	349.5	NaN	NaN	McElhinny [1966]
90	McE_9	-19.70	32.45	1	NaN	164.0	-38.0	NaN	NaN	46.0	336.5	NaN	NaN	McElhinny [1966]

Below is a map reproduced from Gose et al. (2006) that broadly summarizes sampling localities from studies in South Africa and Botswana, including sites from both Pancake and Seidel theses along with some sites from older studies.

```
In [5]: Gose06_Umk_ALL_SITES=Image(filename='Local_PNGs/Umk_all_sites_Gose+06.png')
        display(Gose06_Umk_ALL_SITES)
```



2.4 Data from Seidel (2004)

The W-series sites of Gose et al. (2006) are dolerite sills that intruded the Waterburg Group and are exposed near Middleburg, South Africa. The raw measurement level paleomagnetic data for the W-series specimens are reported in the Master's thesis of Seidel (2004). We digitized all of these data, converted them into MagIC format, and conducted least-squares analysis in order to have directions at the sample level. Having directions at the sample level allows us to calculate means by cooling unit rather than relying on existing site means. The data with all of the fits are imported into a dataframe and displayed as a table. The tilt-corrections used by Seidel (2004) were applied to the data.

```
In [6]: Seidel_dataframe=pd.read_csv('../Data/Prior_Data/Seidel_ALL_FITS.txt',
                                     sep='\t',header=1)
        Seidel_dataframe.head()
```

Out[6]:

	er_analyst_mail_names	er_citation_names	er_location_name	er_sample_name	er_site_name	er_specimen_name	magic_experiment_names
0	NaN	This study	Timbavati	TG01-1	TG01	TG01-1	TG01-1:LP-DIR-AF:LP-DIR-T
1	NaN	This study	Timbavati	TG01-1	TG01	TG01-1	TG01-1:LP-DIR-AF:LP-DIR-T
2	NaN	This study	Timbavati	TG01-1	TG01	TG01-1	TG01-1:LP-DIR-AF:LP-DIR-T
3	NaN	This study	Timbavati	TG01-10	TG01	TG01-10	TG01-10:LP-DIR-AF:LP-DIR-T
4	NaN	This study	Timbavati	TG01-10	TG01	TG01-10	TG01-10:LP-DIR-AF:LP-DIR-T

2.4.1 W-series sites of Seidel (2004)

Geological mapping in the the Middleburg area of South Africa summarized in the simplified map below from Seidel (2004) suggests that these groupings of sites each represent an individual sill: 1. W-01 and W-02 2. W-03, W-06, and W-07 3. W-04 4. W-05 5. W-08 and W-09 6. W-10

We will calculate cooling unit means and virtual geomagnetic poles for each one of these sills. The sites L-1, L-2 and WS-1 are Paleoproterozoic sedimentary rocks that were sampled by Seidel. Two of these sites (L-1 and WS-1) have Umkondo overprints and were included in the overall mean by Gose et al. (2006), but will not be included here as they are not Umkondo cooling units.

```
In [7]: Seidel_Wseries=Image(filename='Local_PNGs/Umk_sites_middleburg_Seidel04.png')
        display(Seidel_Wseries)
```

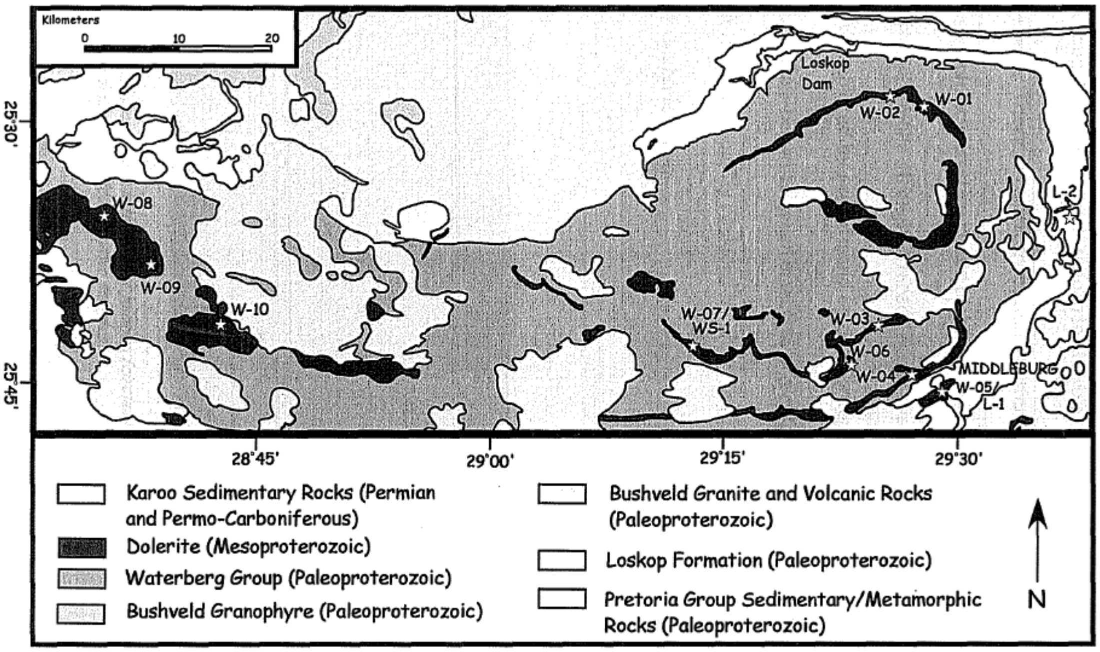


***Figure 17**: Geologic map, modified from Walraven (1978), showing the location of samples taken in the Middleburg area: two Loskop Formation (L) samples, ten post-Waterberg dolerite (W) samples, and one sample from a Waterberg sedimentary unit (WS).*

W01 and W02 Middleburg sill Seidel writes that these two sites within this individual sill are petrographically similar. > "…typical dolerite texture, with both enstatite and clinopyroxene present. There is relatively abundant interstitial microgranophyre ("'15%), and some of the subophitic enstatite shows blebby exsolution." (Seidel, 2004)

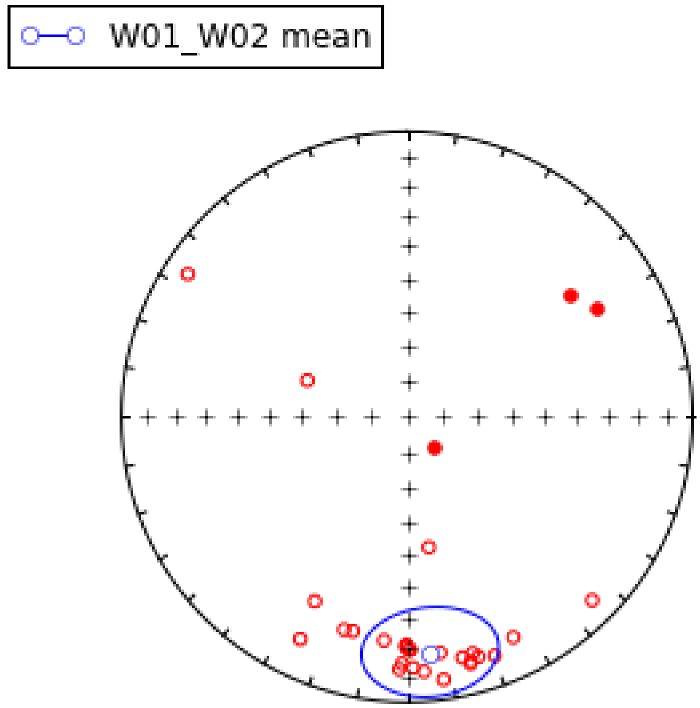
The dataframe of all Seidel (2004) fits can be split by site.

```
In [8]: W01 = Seidel_dataframe.ix[Seidel_dataframe['er_site_name'] == 'W01']
        #slice dataframe to include only tilt corrected fits for W01
        W01_tc = W01[W01['specimen_tilt_correction'] == 100]
        W01_tc.reset_index(inplace=True)

        W02 = Seidel_dataframe.ix[Seidel_dataframe['er_site_name'] == 'W02']
        #slice dataframe to include only tilt corrected fits for W02
        W02_tc = W02[W02['specimen_tilt_correction'] == 100]
        #only include line fits
        W02_tc = W02_tc[W02_tc['specimen_direction_type'] == 'l']
        W02_tc.reset_index(inplace=True)
```

```
In [9]: W01_W02_tc_directions=[]
```

```
#create array of unit vectors from sample fits from sites W01 and W02
for n in range(len(W01_tc)):
    Dec,Inc=W01_tc['specimen_dec'][n],W01_tc['specimen_inc'][n]
    W01_W02_tc_directions.append([Dec,Inc,1.])
for n in range(len(W02_tc)):
    Dec,Inc=W02_tc['specimen_dec'][n],W02_tc['specimen_inc'][n]
    W01_W02_tc_directions.append([Dec,Inc,1.])
#calculate and display fisher mean on EA of all (i.e. unfiltered) directions
W01_W02_tc_mean=pmag.fisher_mean(W01_W02_tc_directions)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(fignum)
IPmag.iplotDI(W01_W02_tc_directions, color='r',label='W01_W02 data')
IPmag.iplotDImean(W01_W02_tc_mean['dec'],W01_W02_tc_mean['inc'],
                  W01_W02_tc_mean["alpha95"],color='b',label='W01_W02 mean')
```

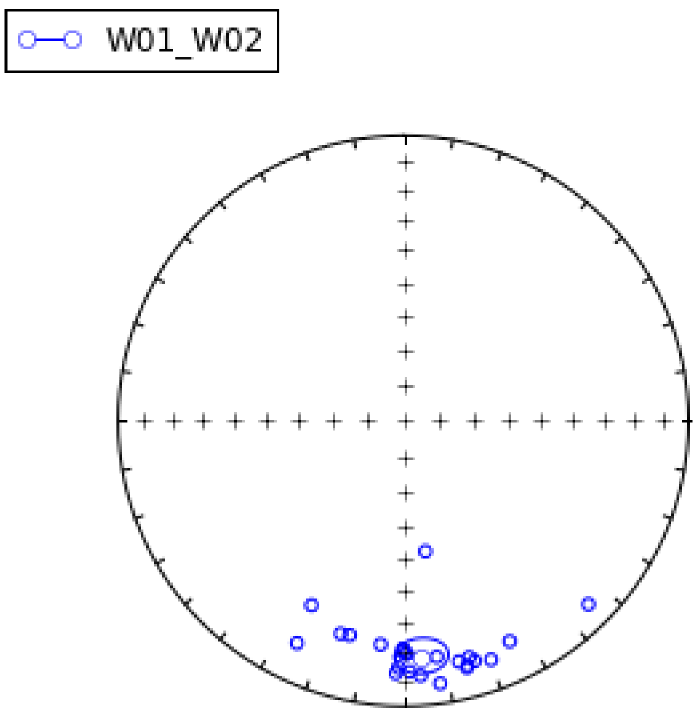
```
/Users/Laurentia/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-packages/matplotlib/axe
  warnings.warn("No labelled objects found. "
```



Some of these data appear to be outliers from the main population. These outliers could be a result of lightning remagnetization or rotated blocks and we take the approach that they should not be included in the calculated mean for the sill. Values that are farther away than 3*α_{95} from the initial mean are excluded from the calculated mean in the code below. A new mean is then calculated.

```
In [10]: #strip out vectors that are greater than a threshold angle
         #defined below as 3*alpha95 here
         W01_W02_tc_directions_edited=[]
         for n in range(len(W01_W02_tc_directions)):
             direction=(W01_W02_tc_directions[n][0],W01_W02_tc_directions[n][1])
             mean_direction=(W01_W02_tc_mean['dec'],W01_W02_tc_mean['inc'])
             threshold_angle=W01_W02_tc_mean['alpha95']*3
             if pmag.angle(direction,mean_direction) < threshold_angle:
                 W01_W02_tc_directions_edited.append([W01_W02_tc_directions[n][0],
                                                      W01_W02_tc_directions[n][1],1.])
```

```
#calculate and plot a new mean for W01/W02 with outliers removed
W01_W02_tc_edited_mean=pmag.fisher_mean(W01_W02_tc_directions_edited)

fignum = 2
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(fignum)
IPmag.iplotDI(W01_W02_tc_directions_edited, color='b')
IPmag.iplotDImean(W01_W02_tc_edited_mean['dec'],W01_W02_tc_edited_mean['inc'],
                  W01_W02_tc_edited_mean["alpha95"],color='b',label='W01_W02')
plt.show()
```




```
In [11]: W01_W02_tc_edited_mean
```

```
Out[11]: {'alpha95': 6.2256544054706602,
          'csd': 17.038008337049813,
```

```
 'dec': 175.55712141560872,
 'inc': -18.376442465316938,
 'k': 22.601246134706432,
 'n': 25,
 'r': 23.938111648492441}
```

W-03, W-06, and W-07 Middleburg sill Site W-03 has a different mean direction than is typical for Umkondo intrusions. Based on this difference, Gose et al. (2006) interpret the sill as being Paleoproterozoic and corresponding to other intrusions pre-dating the Umkondo event that have been grouped into a Waterberg-Soutpansberg Dolorite (WSD) pole. Interestingly, a few samples yielded an Umkondo-like direction, possibly an overprint. However, it should also be noted that there was concern when this site was sampled that the dolerite blocks being sampled might not be in place:

> W-03: Twelve samples were drilled from four blocks embedded in the ground in a level area on the south-facing side of a hill at Fort Merensky. Bedding in the host Waterberg strata above the sampled sill dips 10 to 029. Although the blocks chosen for sampling were quite large, there was considerable concern that some were not in place. Many of the blocks showed evidence of tilting, possibly due to creep. In addition, there was ample evidence that even some of the larger blocks had been moved during construction of the fort. The paleomagnetic analysis indicated that one of the blocks had indeed been moved relative to the others. However, the scatter in the remaining data points (Appendix I) could not be attributed to relative movement of the blocks. (Seidel, 2004)

Sites W-06 and W-07 are from the same intrusion as W-03, and at site W-07 a baked-contact test was performed (WS-1) with inconclusive results. Site W-07 has very scattered demagnetization data not amenable to least-squares fits. W-06 is quite scattered also, but with a few sample yielding directions similiar to those typical for the Umkondo province. Perhaps an Umkondo overprint affected some of the samples from this intrusion. However, another possibility for the variation is that the sample blocks are not demonstrably *in situ*:

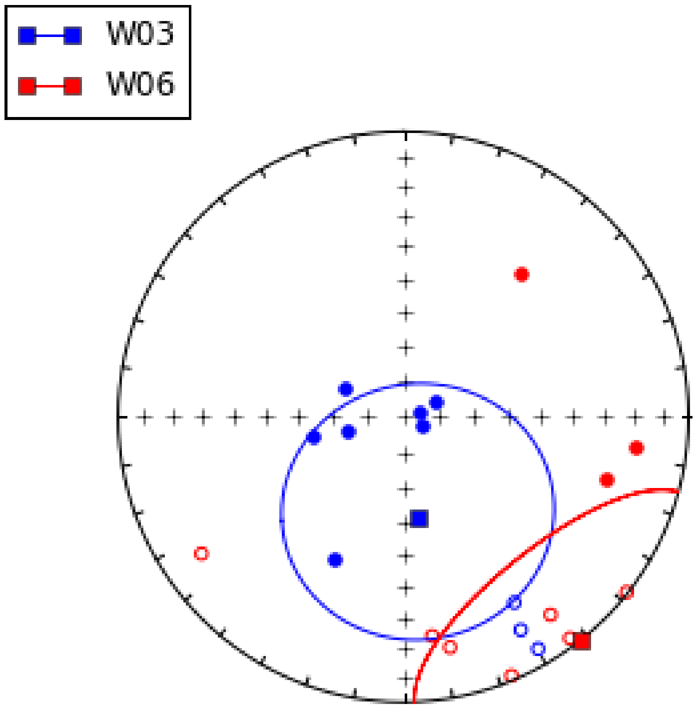
> Twelve samples were drilled from two separate dolerite blocks believed to be in situ. (Seidel, 2004)

Given the Paleoproterozoic age interpretation for the sill, these data are not included in the new Umkondo compilation.

```
In [12]: W03 = Seidel_dataframe.ix[Seidel_dataframe['er_site_name'] == 'W03']
         #slice dataframe so that only tilt corrected fits
         W03_tc = W03[W03['specimen_tilt_correction'] == 100]
         #only include line fits
         W03_tc = W03_tc[W03_tc['specimen_direction_type'] == 'l']
```

```
W03_tc.reset_index(inplace=True)
W06 = Seidel_dataframe.ix[Seidel_dataframe['er_site_name'] == 'W06']
#slice dataframe so that only tilt corrected fits
W06_tc = W06[W06['specimen_tilt_correction'] == 100]
W06_tc.reset_index(inplace=True)
```

In [13]:
```
W03_tc_directions=[]
#create array of unit vectors from sample fits from site W03
for n in range(len(W03_tc)):
    Dec,Inc=W03_tc['specimen_dec'][n],W03_tc['specimen_inc'][n]
    W03_tc_directions.append([Dec,Inc,1.])
#calculate and display fisher mean on an EA of all (i.e. unfiltered) directions
W03_tc_mean=pmag.fisher_mean(W03_tc_directions)

W06_tc_directions=[]
#create array of unit vectors from sample fits from site W06
for n in range(len(W06_tc)):
    Dec,Inc=W06_tc['specimen_dec'][n],W06_tc['specimen_inc'][n]
    W06_tc_directions.append([Dec,Inc,1.])
#calculate and display fisher mean on an EA of all (i.e. unfiltered) directions
W06_tc_mean=pmag.fisher_mean(W06_tc_directions)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(fignum)
IPmag.iplotDI(W03_tc_directions,color='b')
IPmag.iplotDImean(W03_tc_mean['dec'],W03_tc_mean['inc'],W03_tc_mean["alpha95"],
                  color='b',marker='s',label='W03')
IPmag.iplotDI(W06_tc_directions,color='r')
IPmag.iplotDImean(W06_tc_mean['dec'],W06_tc_mean['inc'],W06_tc_mean["alpha95"],
                  color='r',marker='s',label='W06')
```

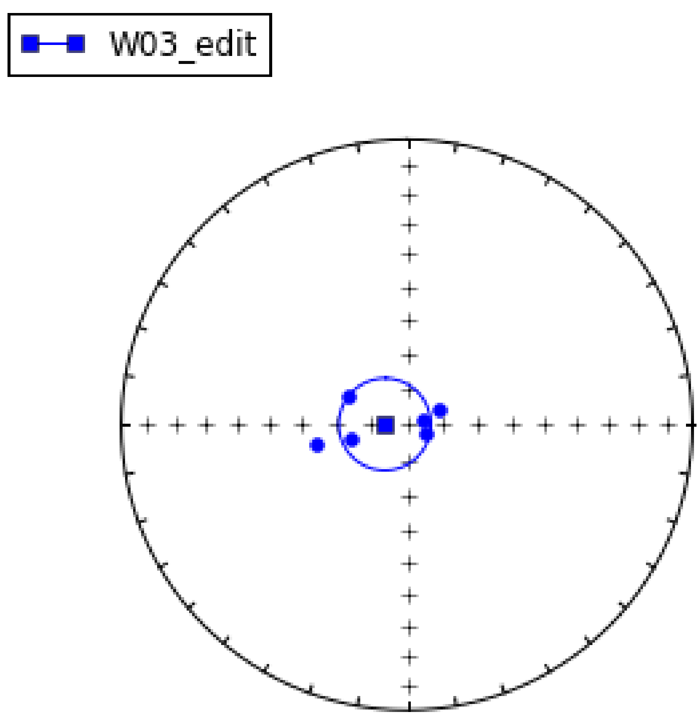


In addition to the steep lower hemisphere direction of the W03 site interpreted by Gose et al. (2006) to be a primary Paleoproterozoic remanence, both the W03 and W06 sites within this intrusion have samples with southerly and shallowly inclined directions that may indicate an Umkondo overprint. However, this intrusion is excluded from the Umkondo group given the Paleoproterozoic interpretation for its age.

W03 appears to have a more consistent mean than W06. We cut out a few scattered data points and include the W03 edited mean in the table of "unknown" directions from older/younger intrusions.

```
In [14]: W03_tc_edit = pd.DataFrame(data=W03_tc)
         W03_tc_edit = W03_tc_edit.drop(1)
         W03_tc_edit = W03_tc_edit.drop(2)
         W03_tc_edit = W03_tc_edit.drop(3)
         W03_tc_edit = W03_tc_edit.drop(4)
         W03_tc_edit.reset_index(inplace=True)
         W03_tc_edit_dir=[]
```

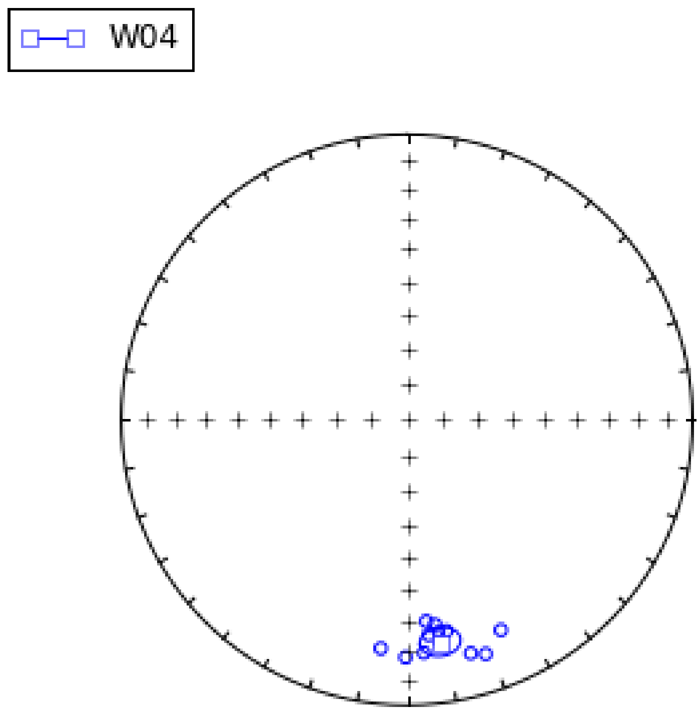
```
#create array of unit vectors from sample fits from site W03
for n in range(len(W03_tc_edit)):
    Dec,Inc=W03_tc_edit['specimen_dec'][n],W03_tc_edit['specimen_inc'][n]
    W03_tc_edit_dir.append([Dec,Inc,1.])
#calculate and display fisher mean on an EA of all (i.e. unfiltered) directions
W03_tc_edit_mean=pmag.fisher_mean(W03_tc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(fignum)
IPmag.iplotDI(W03_tc_edit_dir,color='b')
IPmag.iplotDImean(W03_tc_edit_mean['dec'],W03_tc_edit_mean['inc'],
                  W03_tc_edit_mean["alpha95"],
                  color='b',marker='s',label='W03_edit')
```



W04 sill

W-04: Twelve samples in four clusters were drilled along the banks of the Klein Olifants River, where the river cuts through the dolerite, exposing fresh, in situ blocks (Figure 21). These samples were taken inside the city limits of Middleburg just off Protea Road. Waterberg strata dip 6to 042 downstream of the drill site. No enstatite is present in the sampled dolerite, and interstitial microgranophyre is relatively common. (Seidel, 2004)

In [15]:
```
W04 = Seidel_dataframe.ix[Seidel_dataframe['er_site_name'] == 'W04']
#slice dataframe so that only tilt corrected fits
W04_tc = W04[W04['specimen_tilt_correction'] == 100]
W04_tc.reset_index(inplace=True)

W04_tc_directions=[]
#create array of unit vectors
for n in range(len(W04_tc)):
    Dec,Inc=W04_tc['specimen_dec'][n],W04_tc['specimen_inc'][n]
    W04_tc_directions.append([Dec,Inc,1.])

#calculate and display fisher means on an EA of all (i.e. unfiltered) directions
W04_tc_mean=pmag.fisher_mean(W04_tc_directions)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(fignum)
IPmag.iplotDI(W04_tc_directions, color='b')
IPmag.iplotDImean(W04_tc_mean['dec'],W04_tc_mean['inc'],W04_tc_mean["alpha95"],
                  color='b',marker='s',label='W04')
```



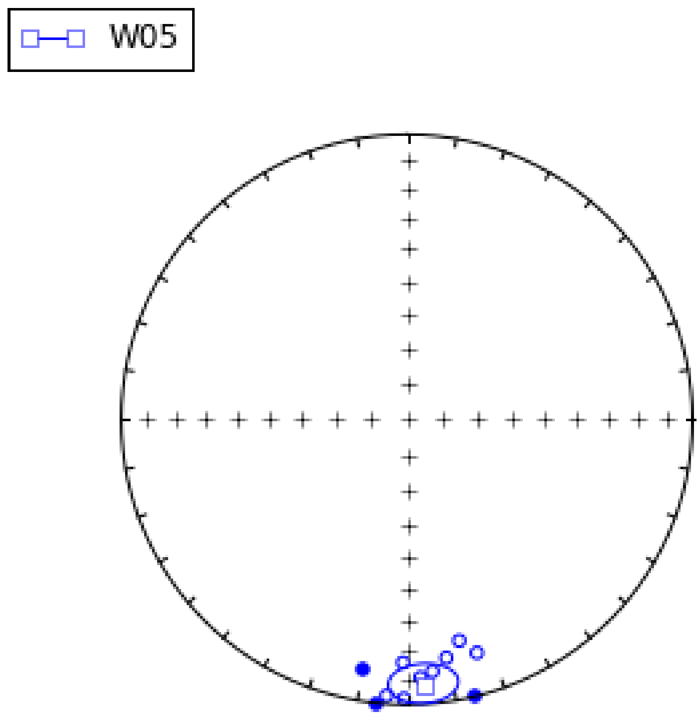

```
In [16]: W04_tc_mean
```

```
Out[16]: {'alpha95': 4.8585230817061573,
          'csd': 9.012449096887476,
          'dec': 171.45131846806996,
          'inc': -22.349249580333709,
          'k': 80.776380340051702,
          'n': 12,
          'r': 11.8638215781186}
```

W05 sill Seidel sampled both the sill (as site W05) and collected samples from host sedimentary rock (site L-1). Gose et al. (2006) included the sill and the overprinted sediment as separate sites each contributing to the overall Umkondo mean. We will include the sill, but not the host rock.

```
In [17]: W05 = Seidel_dataframe.ix[Seidel_dataframe['er_site_name'] == 'W05']
         #slice dataframe so that only tilt corrected fits
         W05_tc = W05[W05['specimen_tilt_correction'] == 100]

         #only include line fits
         W05_tc = W05_tc[W05['specimen_direction_type'] == 'l']
         W05_tc.reset_index(inplace=True)

         W05_tc_directions=[]
         for n in range(len(W05_tc)):
             Dec,Inc=W05_tc['specimen_dec'][n],W05_tc['specimen_inc'][n]
             W05_tc_directions.append([Dec,Inc,1.])
         W05_tc_mean=pmag.fisher_mean(W05_tc_directions)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(fignum)
         IPmag.iplotDI(W05_tc_directions,color='b',label='W05')
         IPmag.iplotDImean(W05_tc_mean['dec'],W05_tc_mean['inc'],W05_tc_mean["alpha95"],
                           color='b',marker='s',label='W05')
```

```
/Users/Laurentia/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-packages/pandas/core/fr
  "DataFrame index.", UserWarning)
```



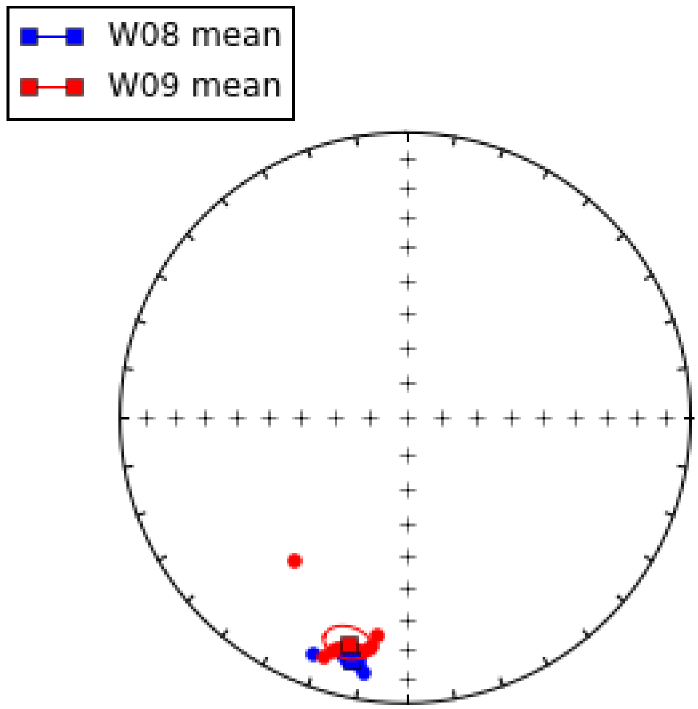

```
In [18]: W05_tc_mean
```

```
Out[18]: {'alpha95': 7.5009902288227179,
          'csd': 13.137343390321529,
          'dec': 176.43536989503451,
          'inc': -7.8215840458300825,
          'k': 38.014994678892293,
          'n': 11,
          'r': 10.736945905570455}
```

W-08 and W-09 sill Data from both sites W-08 and W-09 are very consistent and similar to each other. This is an excellent example of a single intrusion where data should be combined into a single cooling unit mean.

```
In [19]: W08 = Seidel_dataframe.ix[Seidel_dataframe['er_site_name'] == 'W08']
         #slice dataframe so that only tilt corrected fits
         W08_tc = W08[W08['specimen_tilt_correction'] == 100]
         W09 = Seidel_dataframe.ix[Seidel_dataframe['er_site_name'] == 'W09']
         #slice dataframe so that only tilt corrected fits
         W09_tc = W09[W09['specimen_tilt_correction'] == 100]
         W09_tc.reset_index(inplace=True)
         #only include line fits
         W08_tc = W08_tc[W08_tc['specimen_direction_type'] == 'l']
         W08_tc.reset_index(inplace=True)
```

```
In [20]: W08_tc_directions=[]
         W09_tc_directions=[]
         W08_W09_tc_directions=[]
         #create array of unit vectors from sample fits from sites W08 and W09
         for n in range(len(W08_tc)):
             Dec,Inc=W08_tc['specimen_dec'][n],W08_tc['specimen_inc'][n]
             W08_tc_directions.append([Dec,Inc,1.])
             W08_W09_tc_directions.append([Dec,Inc,1.])
         for n in range(len(W09_tc)):
             Dec,Inc=W09_tc['specimen_dec'][n],W09_tc['specimen_inc'][n]
             W09_tc_directions.append([Dec,Inc,1.])
             W08_W09_tc_directions.append([Dec,Inc,1.])
         #calculate and display fisher mean on an EA of all (i.e. unfiltered) directions
         W08_tc_mean=pmag.fisher_mean(W08_tc_directions)
         W09_tc_mean=pmag.fisher_mean(W09_tc_directions)
         W08_W09_tc_mean=pmag.fisher_mean(W08_W09_tc_directions)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(fignum)
         IPmag.iplotDI(W08_tc_directions,'b',label='W08 directions')
         IPmag.iplotDI(W09_tc_directions,'r',label='W09 directions')
         IPmag.iplotDImean(W08_tc_mean['dec'],W08_tc_mean['inc'],W08_tc_mean["alpha95"],
                           marker='s',color='b',label='W08 mean')
         IPmag.iplotDImean(W09_tc_mean['dec'],W09_tc_mean['inc'],W09_tc_mean["alpha95"],
                           marker='s',color='r',label='W09 mean')
         plt.show()
```



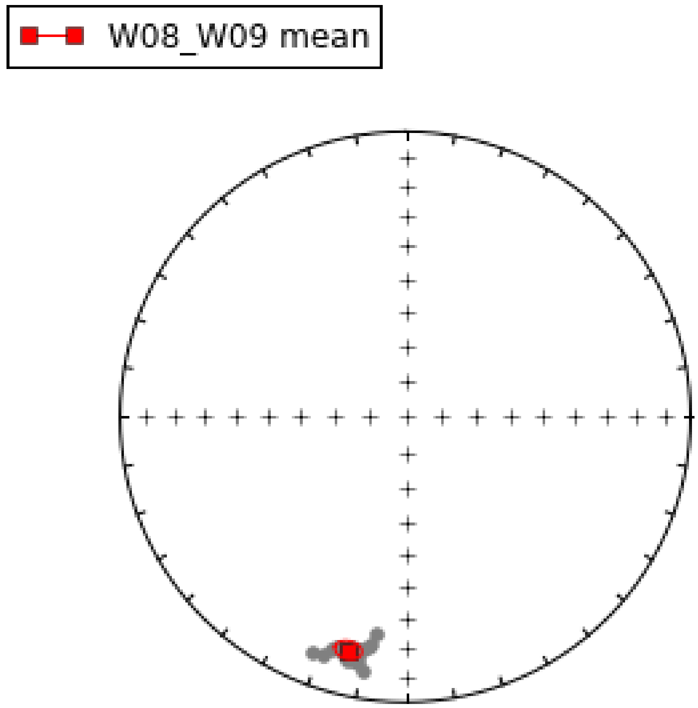
The W08 directions are red while the W09 directions are blue. There is tight correspondance between the data from these two sites which makes sense given that they were collected from the same sill. There appears to be a single point within the W09 directions that is an outlier. Let's remove this point and calculate a mean for the sill that combines the directions from both sites. A test for a common mean is also included below.

In [21]:
```
IPmag.iWatsonV(W08_tc_directions,W09_tc_directions)
```

```
Results of Watson V test:

Watson's V:           6.5
Critical value of V:  6.6
"Pass": Since V is less than Vcrit, the null hypothesis
that the two populations are drawn from distributions
that share a common mean direction can not be rejected.
```

```
M&M1990 classification:

Angle between data set means: 5.8
Critical angle for M&M1990:   5.9
The McFadden and McElhinny (1990) classification for
this test is: 'B'
```

```
In [22]: W08_W09_tc_directions_edited=[]
         for n in range(len(W08_W09_tc_directions)):
             direction=(W08_W09_tc_directions[n][0],W08_W09_tc_directions[n][1])
             mean_direction=(W08_W09_tc_mean['dec'],W08_W09_tc_mean['inc'])
             threshold_angle=W08_W09_tc_mean['alpha95']*5
             if pmag.angle(direction,mean_direction) < threshold_angle:
                 W08_W09_tc_directions_edited.append([W08_W09_tc_directions[n][0],
                                                      W08_W09_tc_directions[n][1],1.])
         #calculate and plot a new mean for W08/W09 with single outlier removed
         W08_W09_tc_edited_mean=pmag.fisher_mean(W08_W09_tc_directions_edited)
         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(fignum)
         IPmag.iplotDI(W08_W09_tc_directions_edited,color='grey')
         IPmag.iplotDImean(W08_W09_tc_mean['dec'],W08_W09_tc_mean['inc'],
                           W08_W09_tc_mean["alpha95"],marker='s',
                           label='W08_W09 mean',color='r')
```

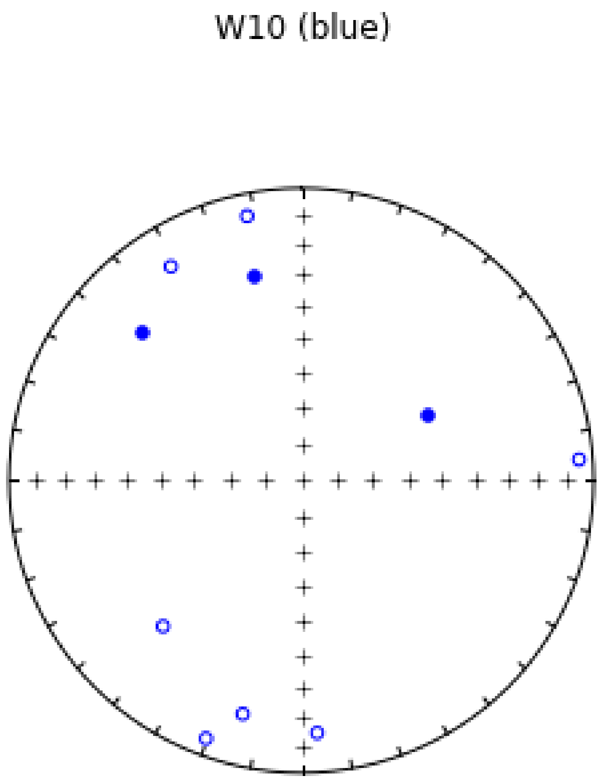



```
In [23]: W08_W09_tc_edited_mean
```

```
Out[23]: {'alpha95': 1.8961539410261299,
          'csd': 4.698138912709072,
          'dec': 192.78872732683382,
          'inc': 15.94213606511043,
          'k': 297.24758194620227,
          'n': 20,
          'r': 19.936080220146454}
```

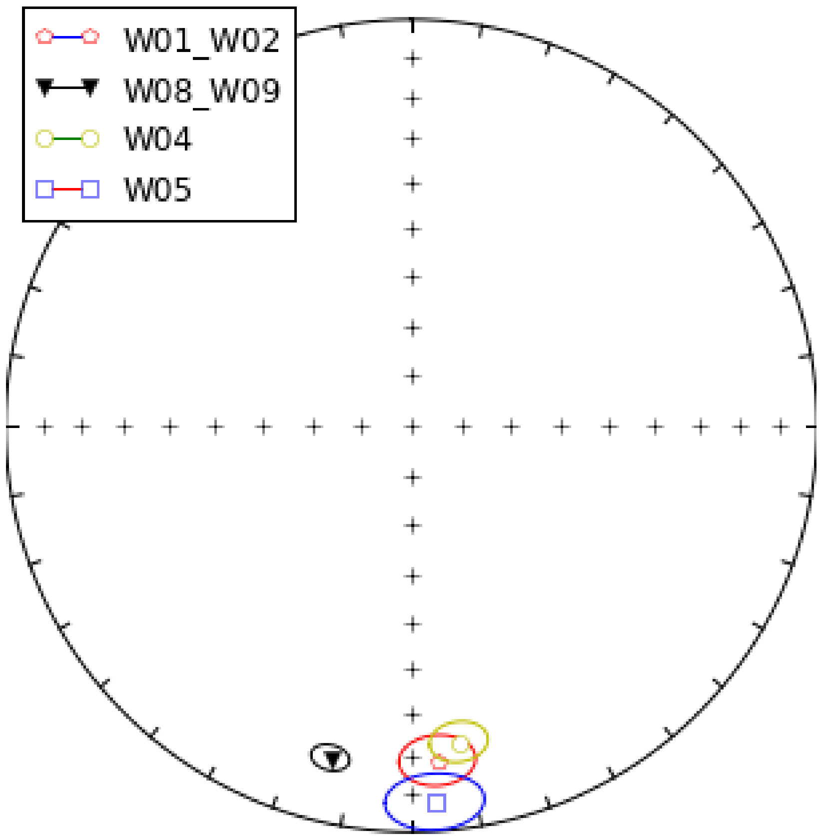
W-10 sill Site W-10 does not yield consistent directions. Some of the samples are near an Umkondo direction, but there is not enough consistency between the directions for a mean direction to be calculated with confidence. This sill was not included in the Gose et al. (2006) compilation and will not be used for mean calculations in this study. Part of the reason for this scatter could be explained by this information provided by Seidel (2004): > W-10: Twelve samples were drilled from four in situ dolerite blocks in a large open field. The low outcrops were visible because of a recent fire.

```
In [24]: W10 = Seidel_dataframe.ix[Seidel_dataframe['er_site_name'] == 'W10']
         #slice dataframe so that only tilt corrected fits
         W10_tc = W10[W10['specimen_tilt_correction'] == 100]
         #only include line fits
         W10_tc = W10_tc[W10['specimen_direction_type'] == 'l']
         W10_tc.reset_index(inplace=True)

         W10_tc_directions=[]
         #create array of unit vectors from sample fits from sites W08 and W09
         for n in range(len(W10_tc)):
             Dec,Inc=W10_tc['specimen_dec'][n],W10_tc['specimen_inc'][n]
             W10_tc_directions.append([Dec,Inc,1.])
         #No need to calculate and plot a mean for this site, it is way too scattered
         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(fignum)
         IPmag.iplotDI(W10_tc_directions,color='b')
         plt.title('W10 (blue)')
         plt.show()
```



Middleburg (W-series sites) area summary from four different cooling units:

```
In [25]: fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(fignum)
         IPmag.iplotDImean(W01_W02_tc_edited_mean['dec'],W01_W02_tc_edited_mean['inc'],
                           W01_W02_tc_edited_mean["alpha95"],marker='p',label='W01_W02',
                           color='r')
         IPmag.iplotDImean(W08_W09_tc_mean['dec'],W08_W09_tc_mean['inc'],
                           W08_W09_tc_mean["alpha95"],marker='v',label='W08_W09',color='k')
         IPmag.iplotDImean(W04_tc_mean['dec'],W04_tc_mean['inc'],
                           W04_tc_mean["alpha95"],marker='o',label='W04',color='y')
         IPmag.iplotDImean(W05_tc_mean['dec'],W05_tc_mean['inc'],
                           W05_tc_mean["alpha95"],marker='s',label='W05',color='b')
```



2.4.2 Add W-series sites to compilation table

```
In [26]: cooling_unit_means = pd.DataFrame(columns=['site_ID','site_lat','site_long',
                             'n','dec_tc','inc_tc','a_95','k','date','date_error'])

In [27]: cooling_unit_means.loc['W01_W02'] = pd.Series({'site_ID':'W01_W02',
                                      'site_lat':-25.48,
                                      'site_long':29.45,
                                      'n':int(W01_W02_tc_edited_mean['n']),
                                      'dec_tc':round(W01_W02_tc_edited_mean['dec'],1),
                                      'inc_tc':round(W01_W02_tc_edited_mean['inc'],1),
                                      'a_95':round(W01_W02_tc_edited_mean['alpha95'],1),
                                      'k':round(W01_W02_tc_edited_mean['k'],1)})

         cooling_unit_means.loc['W04'] = pd.Series({'site_ID':'W04',
```

```
                                    'site_lat':-25.75,
                                    'site_long':29.45,
                                    'n':int(W04_tc_mean['n']),
                                    'dec_tc':round(W04_tc_mean['dec'],1),
                                    'inc_tc':round(W04_tc_mean['inc'],1),
                                    'a_95':round(W04_tc_mean['alpha95'],1),
                                    'k':round(W04_tc_mean['k'],1)})

         cooling_unit_means.loc['W05'] = pd.Series({'site_ID':'W05',
                                    'site_lat':-25.76,
                                    'site_long':29.48,
                                    'n':int(W05_tc_mean['n']),
                                    'dec_tc':round(W05_tc_mean['dec'],1),
                                    'inc_tc':round(W05_tc_mean['inc'],1),
                                    'a_95':round(W05_tc_mean['alpha95'],1),
                                    'k':round(W05_tc_mean['k'],1)})

         cooling_unit_means.loc['W08_W09'] = pd.Series({'site_ID':'W08_W09',
                                    'site_lat':-25.62,
                                    'site_long':29.1,
                                    'n':int(W08_W09_tc_edited_mean['n']),
                                    'dec_tc':round(W08_W09_tc_edited_mean['dec'],1),
                                    'inc_tc':round(W08_W09_tc_edited_mean['inc'],1),
                                    'a_95':round(W08_W09_tc_edited_mean['alpha95'],1),
                                    'k':round(W08_W09_tc_edited_mean['k'],1)})
```

Create another table for all of the directions that are not Umkondo.

```
In [28]: unknown_intrusions = pd.DataFrame(columns=['site_ID','site_lat','site_long',
                              'n','dec_tc','inc_tc','a_95','k','date','date_error'])

         unknown_intrusions.loc['W03'] = pd.Series({'site_ID':'W03',
                                    'site_lat':-25.70,
                                    'site_long':29.41,
                                    'n':int(W03_tc_edit_mean['n']),
                                    'dec_tc':round(W03_tc_edit_mean['dec'],1),
                                    'inc_tc':round(W03_tc_edit_mean['inc'],1),
                                    'a_95':round(W03_tc_edit_mean['alpha95'],1),
                                    'k':round(W03_tc_edit_mean['k'],1)})
```

2.4.3 VF-series sites of Seidel (2004)

Based on the geological maps of the Vredefort area (South Africa), we consider these groupings of sites to each represent an individual sill: 1. VF-1 and VF2 2. VF-3 3. VF-4 and VF5

We calculate cooling unit means and virtual geomagnetic poles for each one of these sills. Only one of these sites (VF-1) was used by Gose et al. (2006) for pole calculations. Combining data, especially in the case of VF-4 and VF-5, yields a better mean.

In [29]:
```
# images for first and second subplots
Seidel_VFseries1 = plt.imread('Local_PNGs/Umk_sites_Vredefort_Seidel04.png')
Seidel_VFseries2 = plt.imread('Local_PNGs/Umk_sites_Vredefort2_Seidel04.png')
# Make figure
fig, (ax1, ax2) = plt.subplots(1,2, figsize=(17,23))
ax1.imshow(Seidel_VFseries1,aspect=None)
ax2.imshow(Seidel_VFseries2,aspect=None)
ax1.set_axis_off() # Hide "spines"
ax2.set_axis_off() # Hide "spines"
```

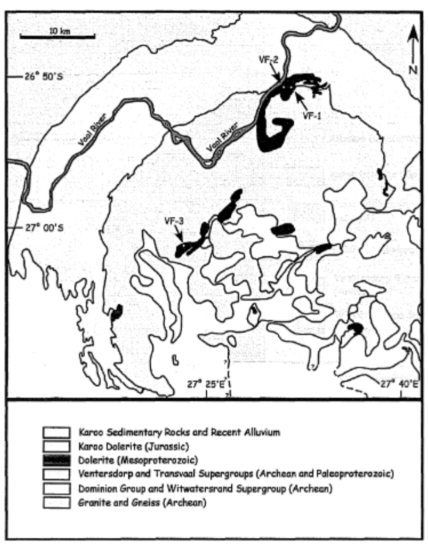


Figure 27: Geologic map showing locations of samples from the dolerite intrusions within the Vredefort impact structure. From Reimold et al. (2000).

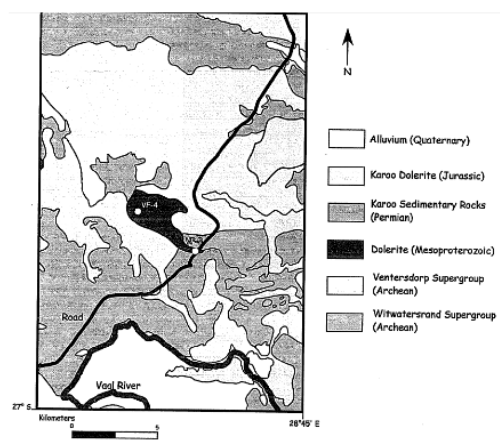
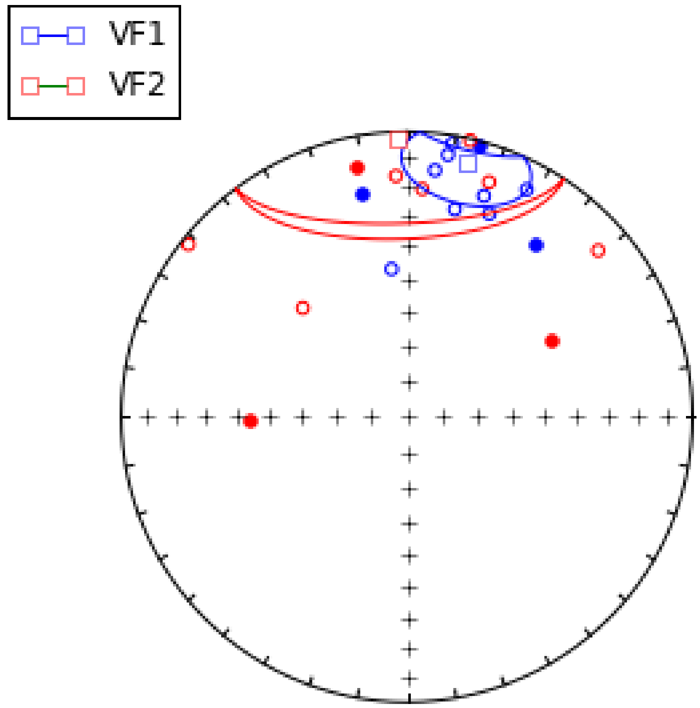


Figure 28: Geologic map showing locations of the two sites at Barnardskop. From Keyser et al. (1986).

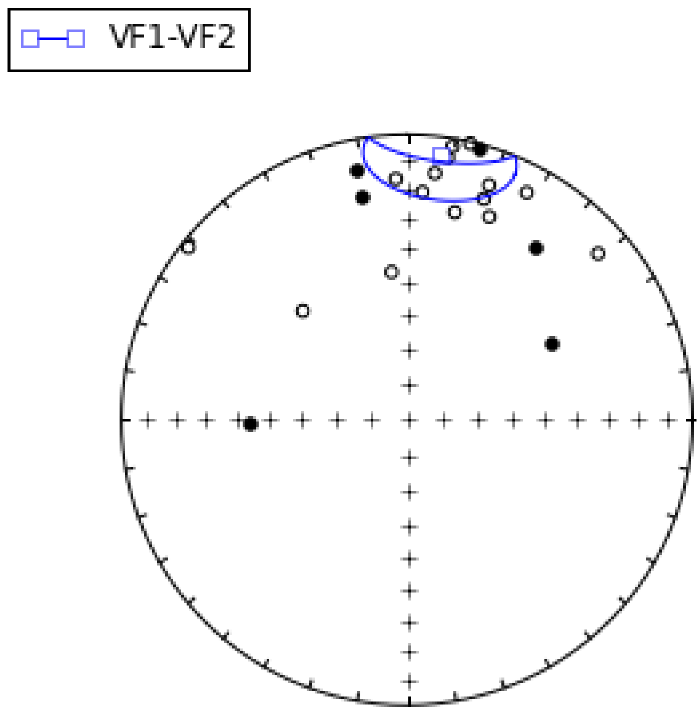
VF-1 and VF-2 intrusion Site VF-1 is the most stable and consistent paleomagnetic site in the VF-series. About half the samples from site VF-2 are consistent with results from VF-1. There was concern at the time of sampling whether the sampled blocks were all *in situ*: >"VF-2: Eleven samples were drilled in three sets from a large outcrop of medium-grained dolerite similar to that at site VF-1. Even though the pieces drilled were relatively large, there was concern that some had experienced rotation either from falling or creep." (Seidel, 2004) A date of 1108.6 ± 1.2 Ma 207Pb/206Pb on baddeleyite) for the VF-2 intrusion (referred to as the Anna's Rust Gabbro) was published in Hanson et al. (2004).

```
In [30]: VF1 = Seidel_dataframe.ix[Seidel_dataframe['er_site_name'] == 'VF1']
         #slice dataframe so that only tilt corrected fits
         VF1_tc = VF1[VF1['specimen_tilt_correction'] == 100]
         #only include line fits
         VF1_tc = VF1_tc[VF1['specimen_direction_type'] == 'l']
         VF1_tc.reset_index(inplace=True)
         VF2 = Seidel_dataframe.ix[Seidel_dataframe['er_site_name'] == 'VF2']
         #slice dataframe so that only tilt corrected fits
         VF2_tc = VF2[VF2['specimen_tilt_correction'] == 100]
         VF2_tc = VF2_tc[VF2['specimen_direction_type'] == 'l']
         VF2_tc.reset_index(inplace=True)
```

```
In [31]: VF1_VF2_tc_directions=[]
         VF1_tc_directions=[]
         #create array of unit vectors from sample fits from sites VF1 and VF2
         for n in range(len(VF1_tc)):
             Dec,Inc=VF1_tc['specimen_dec'][n],VF1_tc['specimen_inc'][n]
             VF1_tc_directions.append([Dec,Inc,1.])
             VF1_VF2_tc_directions.append([Dec,Inc,1.])
         VF2_tc_directions=[]
         for n in range(len(VF2_tc)):
             Dec,Inc=VF2_tc['specimen_dec'][n],VF2_tc['specimen_inc'][n]
             VF2_tc_directions.append([Dec,Inc,1.])
             VF1_VF2_tc_directions.append([Dec,Inc,1.])
         VF1_tc_mean=pmag.fisher_mean(VF1_tc_directions)
         VF2_tc_mean=pmag.fisher_mean(VF2_tc_directions)
         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(VF1_tc_directions,color='b')
         IPmag.iplotDImean(VF1_tc_mean['dec'],VF1_tc_mean['inc'],VF1_tc_mean["alpha95"],
                           color='b',marker='s',label='VF1')
         IPmag.iplotDI(VF2_tc_directions,color='r')
         IPmag.iplotDImean(VF2_tc_mean['dec'],VF2_tc_mean['inc'],VF2_tc_mean["alpha95"],
                           color='r',marker='s',label='VF2')
```

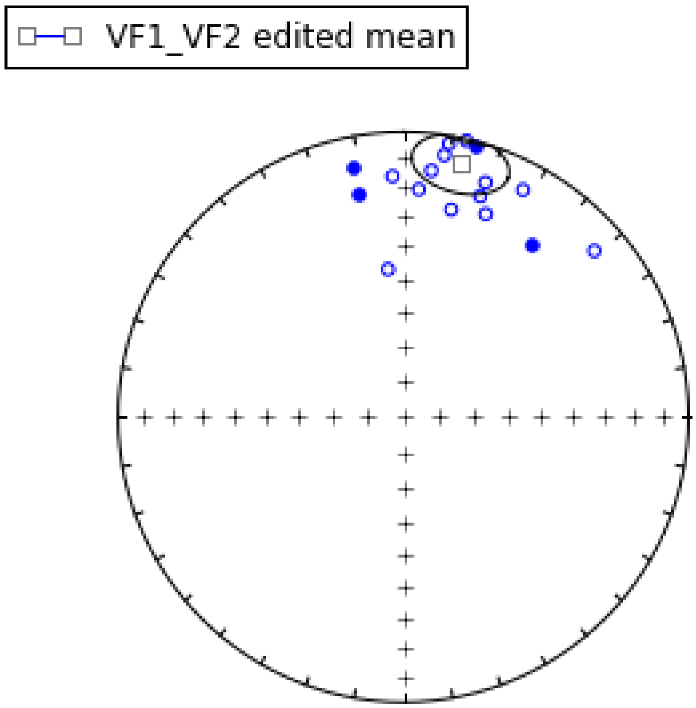


In [32]:
```
#calculate and display fisher mean on an EA of ALL (i.e. unfiltered)...
#...directions for VF1 and VF2
VF1_VF2_tc_mean=pmag.fisher_mean(VF1_VF2_tc_directions)
fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(VF1_VF2_tc_directions,color='black')
IPmag.iplotDImean(VF1_VF2_tc_mean['dec'],VF1_VF2_tc_mean['inc'],
                  VF1_VF2_tc_mean["alpha95"],color='b',marker='s',
                  label='VF1-VF2')
```




```
In [33]: #use the *VF1* mean to exclude outliers that have an angular distance of...
         #...>alpha_95*3 from the VF1 mean and calculate new mean with VF1 and VF2 data
         VF1_VF2_tc_directions_edited=[]
         for n in range(len(VF1_VF2_tc_directions)):
             direction=(VF1_VF2_tc_directions[n][0],VF1_VF2_tc_directions[n][1])
             mean_direction=(VF1_tc_mean['dec'],VF1_tc_mean['inc'])
             threshold_angle=VF1_tc_mean['alpha95']*3
             if pmag.angle(direction,mean_direction) < threshold_angle:
                 VF1_VF2_tc_directions_edited.append([VF1_VF2_tc_directions[n][0],
                                                      VF1_VF2_tc_directions[n][1],1.])
         #calculate and plot a new mean for VF1/VF2 with outliers removed
         VF1_VF2_tc_edited_mean=pmag.fisher_mean(VF1_VF2_tc_directions_edited)
         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(fignum)
```

```
IPmag.iplotDI(VF1_VF2_tc_directions_edited,color='b')
IPmag.iplotDImean(VF1_VF2_tc_edited_mean['dec'],VF1_VF2_tc_edited_mean['inc'],
                  VF1_VF2_tc_edited_mean["alpha95"],marker='s',
                  label='VF1_VF2 edited mean')
```



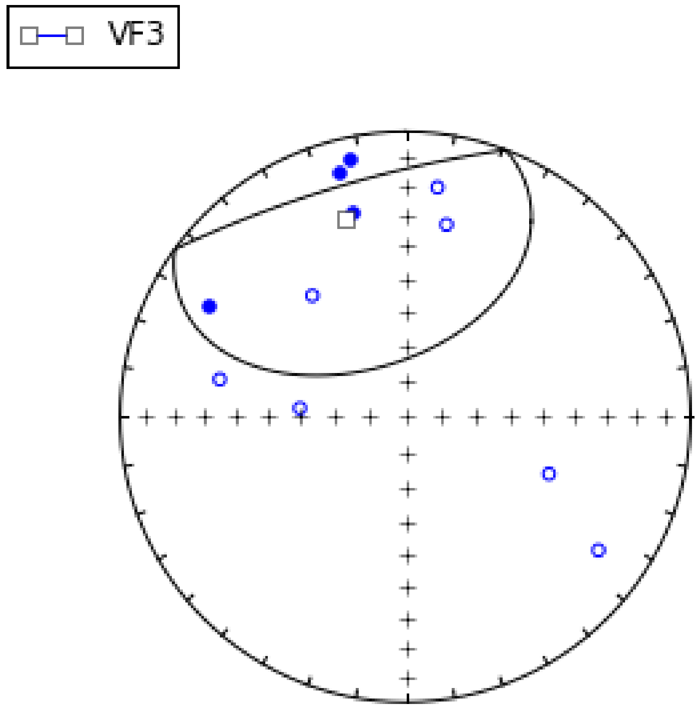

```
In [34]: VF1_VF2_tc_edited_mean
```

```
Out[34]: {'alpha95': 10.786838933965125,
          'csd': 23.471977060138816,
          'dec': 12.764349138882814,
          'inc': -9.5137094637782216,
          'k': 11.908873817862052,
          'n': 17,
          'r': 15.656464058255308}
```

VF-3 No other sites were sampled in the same intrusion as VF-3. This intrusion was excluded from the calculations of Gose et al. (2006) as a result of the paleomagnetic results being scattered as can be seen in the equal area plot below. The intrusion is reported to be slightly altered by Seidel (see below), including olivine grains that yielded fine-grained magnetite as an alteration product. Given the lack of consistency between the sample directions, this site will not be included in the grand mean calculated here.

> The pyroxene is lightly altered to actinolite and smectite, and the plagioclase is lightly sericitized. Anhedral olivine (~10%) is surrounded by pyroxene reaction rims. No interstitial microgranophyre is present. The olivine is partly altered to bastite and very fine-grained magnetite. (Seidel, 2004)

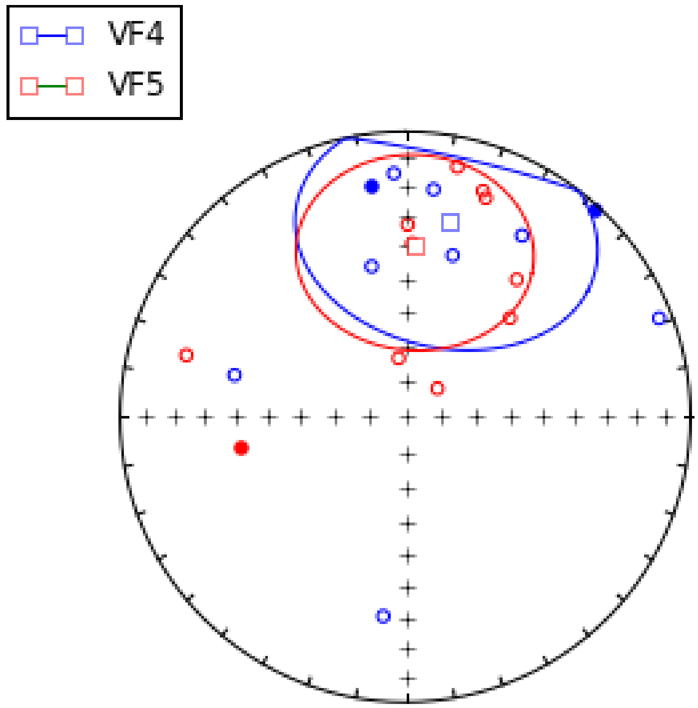
```
In [35]: VF3 = Seidel_dataframe.ix[Seidel_dataframe['er_site_name'] == 'VF3']
         #slice dataframe to include only tilt corrected fits for VF3
         VF3_tc = VF3[VF3['specimen_tilt_correction'] == 100]
         VF3_tc.reset_index(inplace=True)
         VF3_tc_directions=[]
         #create array of unit vectors from sample fits from sites VF1 and VF2
         for n in range(len(VF1_tc)):
             Dec,Inc=VF3_tc['specimen_dec'][n],VF3_tc['specimen_inc'][n]
             VF3_tc_directions.append([Dec,Inc,1.])
         VF3_tc_mean=pmag.fisher_mean(VF3_tc_directions)
         #Excluded plot of data...
         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(VF3_tc_directions,color='b')
         IPmag.iplotDImean(VF3_tc_mean['dec'],VF3_tc_mean['inc'],VF3_tc_mean["alpha95"],
                           color='black',marker='s',label='VF3')
```



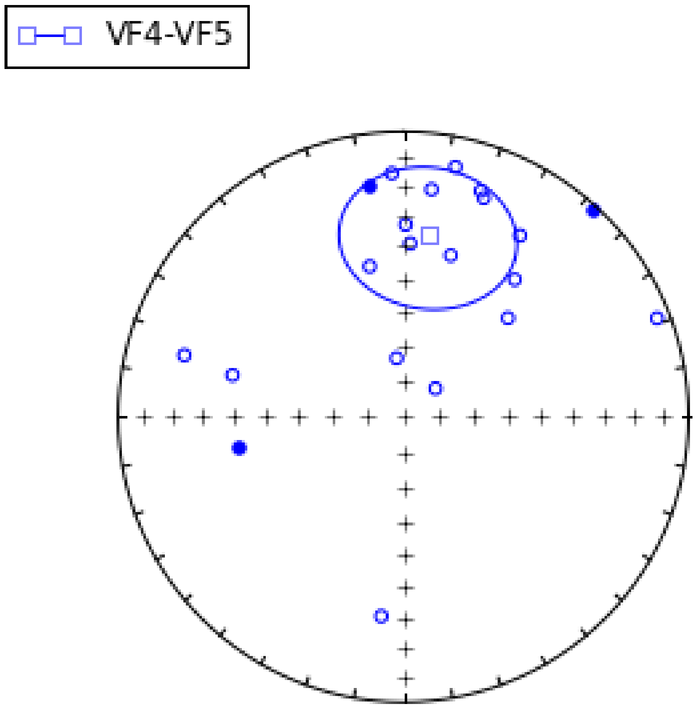
VF-4 and VF-5 Both sites VF-4 and VF-5 were excluded from calculations of Gose et al. (2006), but when combined they yield a better constrained mean (given that they are from the same sill).

```
In [36]: VF4 = Seidel_dataframe.ix[Seidel_dataframe['er_site_name'] == 'VF4']
         #slice dataframe so that only tilt corrected fits
         VF4_tc = VF4[VF4['specimen_tilt_correction'] == 100]
         #only include line fits
         VF4_tc = VF4_tc[VF4['specimen_direction_type'] == 'l']
         VF4_tc.reset_index(inplace=True)
         VF5 = Seidel_dataframe.ix[Seidel_dataframe['er_site_name'] == 'VF5']
         #slice dataframe so that only tilt corrected fits
         VF5_tc = VF5[VF5['specimen_tilt_correction'] == 100]
         VF5_tc = VF5_tc[VF5['specimen_direction_type'] == 'l']
         VF5_tc.reset_index(inplace=True)
```

In [37]:
```
VF4_VF5_tc_directions=[]
VF4_tc_directions=[]
#create array of unit vectors from sample fits from sites VF4 and VF5
for n in range(len(VF4_tc)):
    Dec,Inc=VF4_tc['specimen_dec'][n],VF4_tc['specimen_inc'][n],
    VF4_tc_directions.append([Dec,Inc,1.])
    VF4_VF5_tc_directions.append([Dec,Inc,1.])
VF5_tc_directions=[]
for n in range(len(VF5_tc)):
    Dec,Inc=VF5_tc['specimen_dec'][n],VF5_tc['specimen_inc'][n],
    VF5_tc_directions.append([Dec,Inc,1.])
    VF4_VF5_tc_directions.append([Dec,Inc,1.])
VF4_tc_mean=pmag.fisher_mean(VF4_tc_directions)
VF5_tc_mean=pmag.fisher_mean(VF5_tc_directions)
fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(VF4_tc_directions,color='b')
IPmag.iplotDImean(VF4_tc_mean['dec'],VF4_tc_mean['inc'],
                  VF4_tc_mean["alpha95"],color='b',marker='s',label='VF4')
IPmag.iplotDI(VF5_tc_directions,color='r')
IPmag.iplotDImean(VF5_tc_mean['dec'],VF5_tc_mean['inc'],
                  VF5_tc_mean["alpha95"],color='r',marker='s',label='VF5')
```

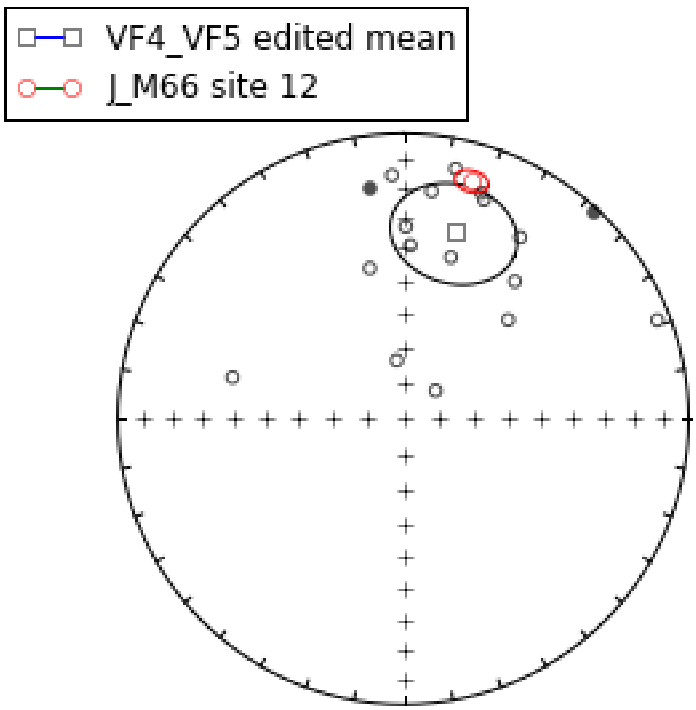


In [38]:
```
#calculate/display fisher mean on an EA of ALL (i.e. unfiltered)...
#...directions for VF4 and VF5
VF4_VF5_tc_mean=pmag.fisher_mean(VF4_VF5_tc_directions)
fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(VF4_VF5_tc_directions,color='b')
IPmag.iplotDImean(VF4_VF5_tc_mean['dec'],VF4_VF5_tc_mean['inc'],
                  VF4_VF5_tc_mean["alpha95"],
                  color='b',marker='s',label='VF4-VF5')
plt.show()
```



In [39]:
```
VF4_VF5_tc_directions_edited = []
for n in range(len(VF4_VF5_tc_directions)):
    direction=(VF4_VF5_tc_directions[n][0],VF4_VF5_tc_directions[n][1])
    mean_direction=(VF4_VF5_tc_mean['dec'],VF4_VF5_tc_mean['inc'])
    threshold_angle=VF4_VF5_tc_mean['alpha95']*3
    if pmag.angle(direction,mean_direction) < threshold_angle:
        VF4_VF5_tc_directions_edited.append([VF4_VF5_tc_directions[n][0],
                                             VF4_VF5_tc_directions[n][1],1.])
#calculate and plot a new mean for VF4 and VF5 with outliers removed
VF4_VF5_tc_edited_mean=pmag.fisher_mean(VF4_VF5_tc_directions_edited)
fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(fignum)
IPmag.iplotDI(VF4_VF5_tc_directions_edited,color='0.3')
IPmag.iplotDImean(VF4_VF5_tc_edited_mean['dec'],VF4_VF5_tc_edited_mean['inc'],
```

```
                   VF4_VF5_tc_edited_mean["alpha95"],marker='s',
                   label='VF4_VF5 edited mean')
IPmag.iplotDImean(16,-14.5,3.9,color='r',label='J_M66 site 12')
plt.show()
```

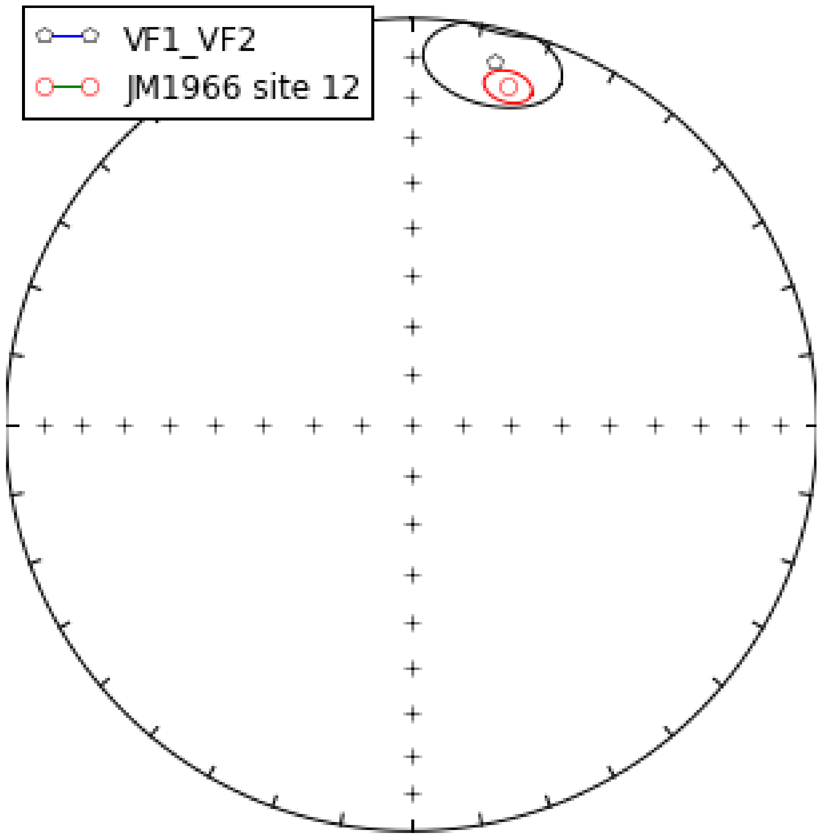


The Jones and McElhinny site mean from this same sill (12) is much more tightly constrained and we include it in the compilation rather than the VF4_VF5 data. We would combine all the results to create one mean, but we do not have the sample level data from Jones and McElhinny. Because the VF4 and VF5 data are more scattered than, but consistent with, 'site 12' from Jones and McElhinny, we choose to use the Jones and McElhinny results (they are added to the compilation later in this analysis). Hanson et al. (2004) published a date for VF4, 1108.5 0.8 Ma.

Vredefort (VF-series sites) area summary from two different cooling units

```
In [40]: fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
```

```
IPmag.iplotNET(fignum)
IPmag.iplotDImean(VF1_VF2_tc_edited_mean['dec'],VF1_VF2_tc_edited_mean['inc'],
                  VF1_VF2_tc_edited_mean["alpha95"],marker='p',label='VF1_VF2')
IPmag.iplotDImean(16,-14.5,3.9,color='r',label='JM1966 site 12')
plt.show()
```




```
In [41]: cooling_unit_means.loc['VF1_VF2'] = pd.Series({'site_ID':'VF1_VF2',
                                             'site_lat':-25.8,
                                             'site_long':27.5,
                                             'n':int(VF1_VF2_tc_mean['n']),
                                             'dec_tc':round(VF1_VF2_tc_mean['dec'],1),
                                             'inc_tc':round(VF1_VF2_tc_mean['inc'],1),
                                             'a_95':round(W08_W09_tc_mean['alpha95'],1),
                                             'k':round(W08_W09_tc_mean['k'],1),
                                             'date':1108.6,
                                             'date_error':1.2})
```

2.4.4 Timbavati Intrusion (TG-series) of Seidel (2004)

The Timbavati Gabbro is an intrusive body that extends for over 300 km near the border of Mozambique. The southern portion of the sill near TG-03 reaches thicknesses of over 400 m, and near site TG-05 has a thickness of ~200 m (Walraven, 1986). The data compilation of Gose et al. (2006) calculated means for geographically distinct sites within the Timbavati Gabbro and then calculated a mean of these sites to use as one of the area means that was assigned unit vector weight in the overall mean Umkondo pole.

In contrast to the dolerite sills discussed thus far, determining what constitutes an individual cooling unit within the Timbavati Gabbro is difficult. Geochemical data developed by Bullen (2005) identified distinct geochemical groups within individual mapped sills indicating that the bodies are composite intrusions formed by multiple pulses of magma. Distinct petrology of olivine gabbro, gabbro and quartz gabbro are present, but have been interpretted to be interlayered and consanguineous (Walraven, 1986; Hargraves et al., 1994). Delineating distinct cooling units is therefore difficult. If the approach is taken to consider the gabbro as a single body, a sheet-like intrusion with a thickness of 200 m (applicable to a half-space cooling model, see Delaney (1987)), that intrudes at approximately 1150°C into a host rock of 50°C could take ~800 yrs to cool below 500°C at its center. The margin of the intrusion would have cooled significantly faster than the center of the intrusion. It is unclear how the different TG-series sites are distributed with respect to the margins, but many appear to be from coarse-grained regions. Even with all sites having been taken from the interior, the timescale of cooling is not long enough to fully average out secular variation.

One division that does seem clear is that the data from Bullen (2005) revealed a unique geochemical signature throughout the northernmost exposure (corresponding to sites TG-08 and TG-09) suggesting that it is distinct from the intrusive rocks to the south. Divisions between the rest of the sites are difficult. We tentatively take the approach of considering the rest of the sites as being from a single large cooling unit and group sites TG-01, TG-02, TG-03, TG-04, TG-05, TG-06, TG-07, TG-11 and TG-12 as being from the same thick sill. This approach is a crude approximation that could be refined through subsequent work.

```
In [42]: Seidel_TGseries=Image(filename=
                                   'Local_PNGs/Umk_sites_Timbavati_gabbro_Seidel04.png')
         display(Seidel_TGseries)
```

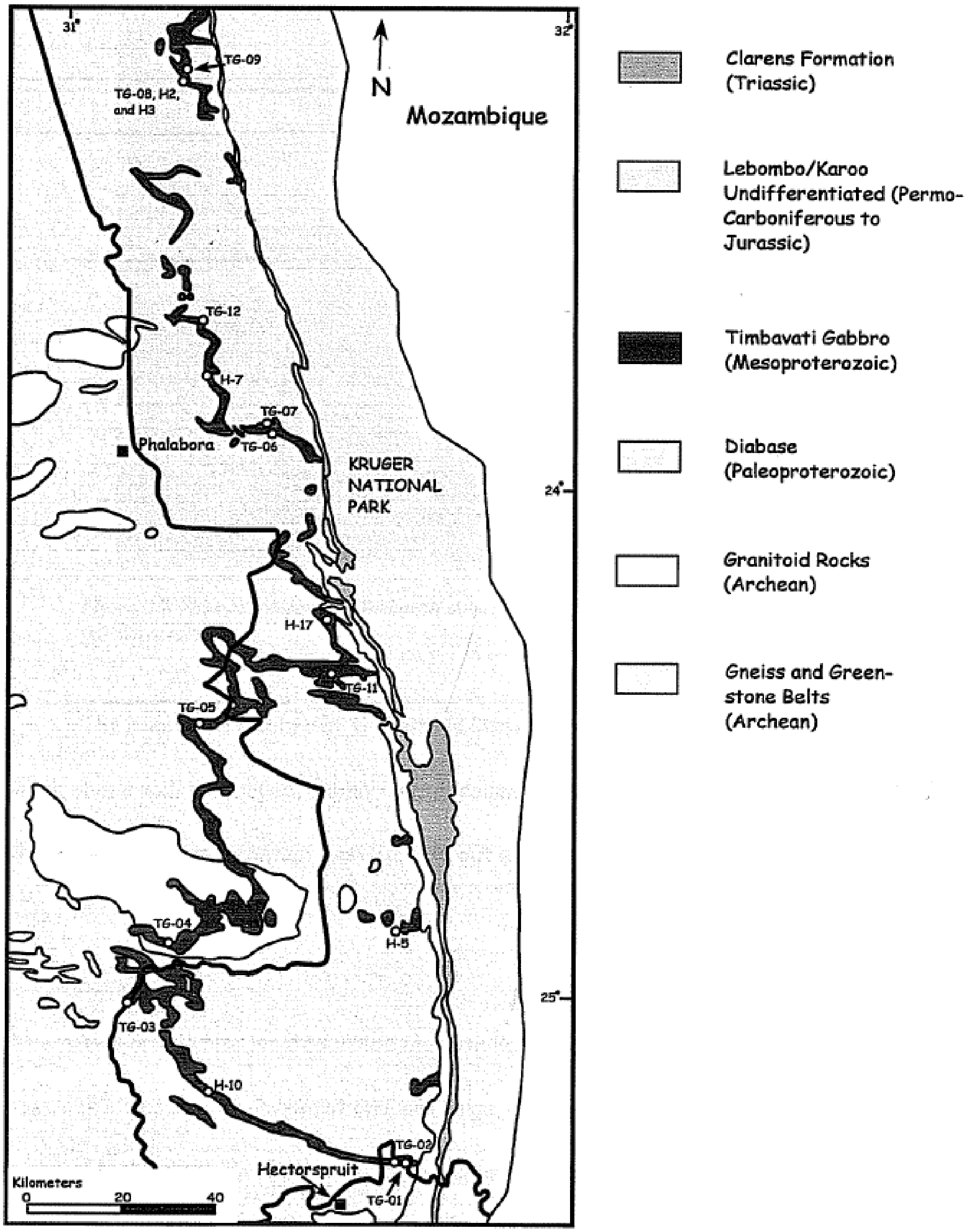


Figure 9: Geologic map showing locations of samples taken from the Timbavati Gabbro (TG), modified from Keyser (1997). Also included are samples taken by Hargraves et al. (1994), indicated by "H" prefix.

```
In [43]: TG = Seidel_dataframe.loc[Seidel_dataframe['er_location_name'] == 'Timbavati']
         #slice dataframe so that only tilt corrected fits
         TG_tc = TG[TG['specimen_tilt_correction'] == 100]
         #only include line fits
         TG_tc = TG_tc[TG_tc['specimen_direction_type'] == 'l']
         TG_tc.reset_index(inplace=True,drop=True)
         TG_tc.head()
```

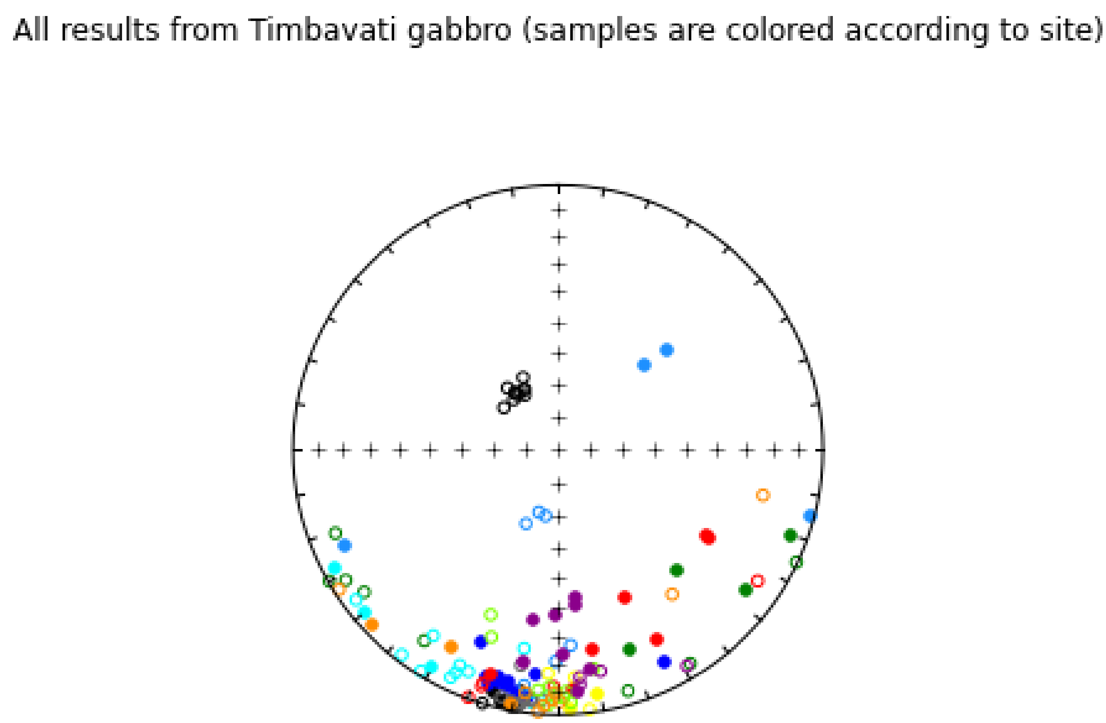
Out[43]:

	er_analyst_mail_names	er_citation_names	er_location_name	er_sample_name	er_site_name	er_specimen_name	magic_experiment_names	m
0	NaN	This study	Timbavati	TG01-1	TG01	TG01-1	TG01-1:LP-DIR-AF:LP-DIR-T	
1	NaN	This study	Timbavati	TG01-10	TG01	TG01-10	TG01-10:LP-DIR-AF:LP-DIR-T	
2	NaN	This study	Timbavati	TG01-11	TG01	TG01-11	TG01-11:LP-DIR-AF:LP-DIR-T	
3	NaN	This study	Timbavati	TG01-12	TG01	TG01-12	TG01-12:LP-DIR-AF:LP-DIR-T	
4	NaN	This study	Timbavati	TG01-13	TG01	TG01-13	TG01-13:LP-DIR-AF:LP-DIR-T	

```
In [44]: TG_08 = TG_tc.loc[TG_tc['er_site_name'] == 'TG08']
         TG_08.reset_index(inplace=True)
         TG_09 = TG_tc.loc[TG_tc['er_site_name'] == 'TG09']
         TG_09.reset_index(inplace=True)

         #seperate sites out so that they can be plotted in different colors
         TG_01 = TG_tc.loc[TG_tc['er_site_name'] == 'TG01']
         TG_01.reset_index(inplace=True)
         TG_02 = TG_tc.loc[TG_tc['er_site_name'] == 'TG02']
         TG_02.reset_index(inplace=True)
         TG_03 = TG_tc.loc[TG_tc['er_site_name'] == 'TG03']
         TG_03.reset_index(inplace=True)
         TG_04 = TG_tc.loc[TG_tc['er_site_name'] == 'TG04']
         TG_04.reset_index(inplace=True)
         TG_05 = TG_tc.loc[TG_tc['er_site_name'] == 'TG05']
         TG_05.reset_index(inplace=True)
         TG_06 = TG_tc.loc[TG_tc['er_site_name'] == 'TG06']
         TG_06.reset_index(inplace=True)
         TG_07 = TG_tc.loc[TG_tc['er_site_name'] == 'TG07']
         TG_07.reset_index(inplace=True)
         TG_11 = TG_tc.loc[TG_tc['er_site_name'] == 'TG11']
         TG_11.reset_index(inplace=True)
         TG_12 = TG_tc.loc[TG_tc['er_site_name'] == 'TG12']
         TG_12.reset_index(inplace=True)

         TG_01_directions=[]
         for n in range(len(TG_01)):
             Dec,Inc=TG_01['specimen_dec'][n],TG_01['specimen_inc'][n]
             TG_01_directions.append([Dec,Inc,1.])
         TG_01_mean=pmag.fisher_mean(TG_01_directions)
```

```
TG_02_directions=[]
for n in range(len(TG_02)):
    Dec,Inc=TG_02['specimen_dec'][n],TG_02['specimen_inc'][n]
    TG_02_directions.append([Dec,Inc,1.])
TG_02_mean=pmag.fisher_mean(TG_02_directions)
TG_03_directions=[]
for n in range(len(TG_03)):
    Dec,Inc=TG_03['specimen_dec'][n],TG_03['specimen_inc'][n]
    TG_03_directions.append([Dec,Inc,1.])
TG_03_mean=pmag.fisher_mean(TG_03_directions)
TG_04_directions=[]
for n in range(len(TG_04)):
    Dec,Inc=TG_04['specimen_dec'][n],TG_04['specimen_inc'][n]
    TG_04_directions.append([Dec,Inc,1.])
TG_04_mean=pmag.fisher_mean(TG_04_directions)
TG_05_directions=[]
for n in range(len(TG_05)):
    Dec,Inc=TG_05['specimen_dec'][n],TG_05['specimen_inc'][n]
    TG_05_directions.append([Dec,Inc,1.])
TG_05_mean=pmag.fisher_mean(TG_05_directions)
TG_06_directions=[]
for n in range(len(TG_06)):
    Dec,Inc=TG_06['specimen_dec'][n],TG_06['specimen_inc'][n]
    TG_06_directions.append([Dec,Inc,1.])
TG_06_mean=pmag.fisher_mean(TG_06_directions)
TG_07_directions=[]
for n in range(len(TG_07)):
    Dec,Inc=TG_07['specimen_dec'][n],TG_07['specimen_inc'][n]
    TG_07_directions.append([Dec,Inc,1.])
TG_07_mean=pmag.fisher_mean(TG_07_directions)
TG_08_directions=[]
for n in range(len(TG_08)):
    Dec,Inc=TG_08['specimen_dec'][n],TG_08['specimen_inc'][n]
    TG_08_directions.append([Dec,Inc,1.])
TG_08_mean=pmag.fisher_mean(TG_08_directions)
TG_09_directions=[]
for n in range(len(TG_09)):
    Dec,Inc=TG_09['specimen_dec'][n],TG_09['specimen_inc'][n]
    TG_09_directions.append([Dec,Inc,1.])
TG_09_mean=pmag.fisher_mean(TG_09_directions)
TG_11_directions=[]
for n in range(len(TG_11)):
    Dec,Inc=TG_11['specimen_dec'][n],TG_11['specimen_inc'][n]
    TG_11_directions.append([Dec,Inc,1.])
TG_11_mean=pmag.fisher_mean(TG_11_directions)
```

```
TG_12_directions=[]
for n in range(len(TG_12)):
    Dec,Inc=TG_12['specimen_dec'][n],TG_12['specimen_inc'][n]
    TG_12_directions.append([Dec,Inc,1.])
TG_12_mean=pmag.fisher_mean(TG_12_directions)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(TG_01_directions,color='black',label='TG01')
IPmag.iplotDI(TG_02_directions,color='blue',label='TG02')
IPmag.iplotDI(TG_03_directions,color='green',label='TG03')
IPmag.iplotDI(TG_04_directions,color='red',label='TG04')
IPmag.iplotDI(TG_05_directions,color='cyan',label='TG05')
IPmag.iplotDI(TG_06_directions,color='dimgray',label='TG06')
IPmag.iplotDI(TG_07_directions,color='yellow',label='TG07')
IPmag.iplotDI(TG_08_directions,color='chartreuse',label='TG08')
IPmag.iplotDI(TG_09_directions,color='dodgerblue',label='TG09')
IPmag.iplotDI(TG_11_directions,color='darkorange',label='TG11')
IPmag.iplotDI(TG_12_directions,color='darkmagenta',label='TG12')
plt.title('All results from Timbavati gabbro (samples are colored according to site)')
plt.show()
```

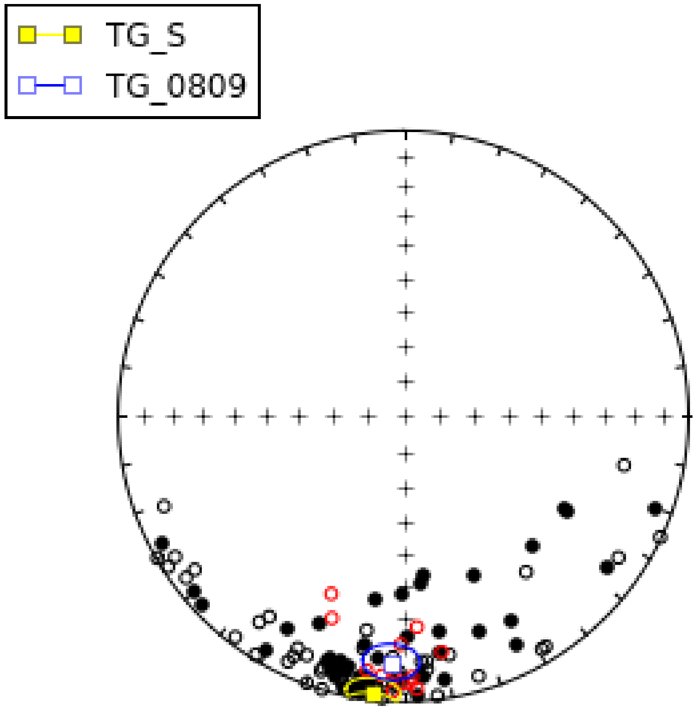


TG01 is a baked-contact test for a Karoo dike (the tight grouping of vectors near [330,-80]) into an Umkondo Sill, so half the samples are the baked Karoo direction. The baked-contact test (BCT) results represent a positive inverse baked-contact test, confirming that the magnetization from the Timbavati gabbro is older then the Karoo LIP in age. An additional note on the BCT: the original authors suggest that the baked zone is asymmetrical, but cover on the margins of the dike allow enough wiggle room for the baked zones to still be symmetrical in width.

TG03 samples (dark green) and TG04 are both quite scattered. The rest of the sites appear to have at least some samples that yield Umkondo-like directions.

```
In [45]: #cut out Karoo baked components using dec;
         #they are only components with dec btw 270 and 360
         TG_tc = TG_tc[TG_tc['specimen_dec'] < 270]
         TG_tc.reset_index(inplace=True)
         #create a unit vector array for TG-series sites
         TG_tc_directions=[]
         for n in range(len(TG_tc)):
             Dec,Inc=TG_tc['specimen_dec'][n],TG_tc['specimen_inc'][n]
```

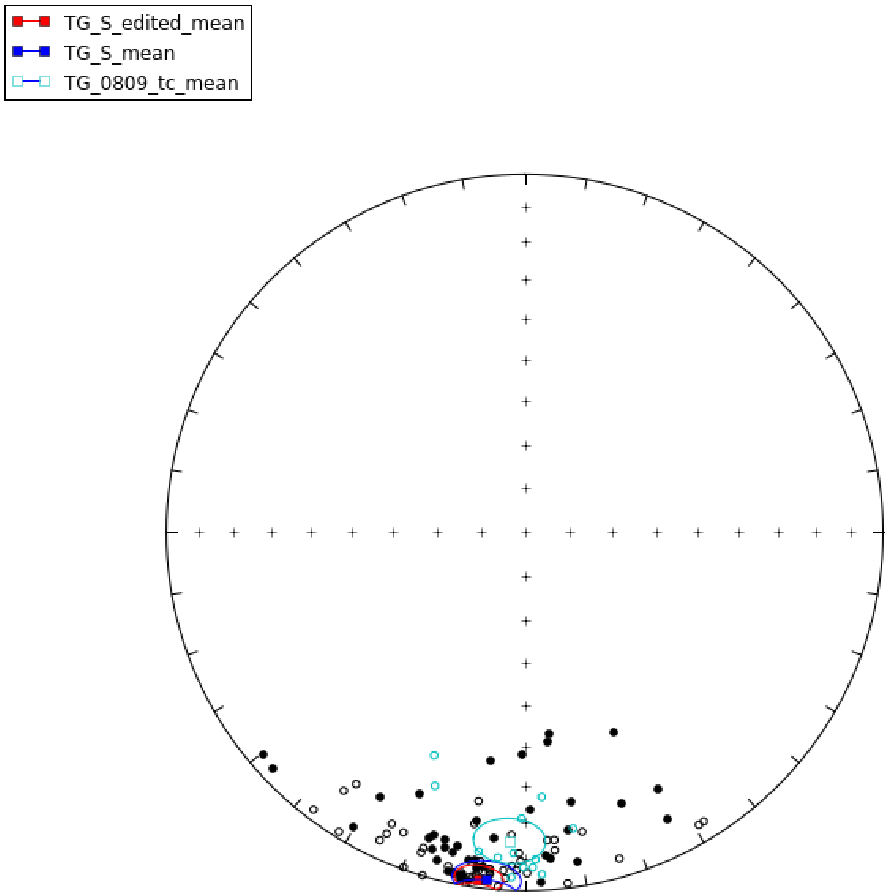
```
    TG_tc_directions.append([Dec,Inc,1.])
TG_tc_mean=pmag.fisher_mean(TG_tc_directions)

#Make dataframe with all data besided TG08 and TG09,...
#...abbreviation TG_S referes to "southern" sites
TG_S = TG_tc
TG_S = TG_S.loc[TG_S['er_site_name'] != 'TG08']
TG_S = TG_S.loc[TG_S['er_site_name'] != 'TG09']
TG_S.reset_index(inplace=True, drop=True)
#create unit vector array for TG-series S sites, excluding TG08 and TG09
TG_S_directions=[]
for n in range(len(TG_S)):
    Dec,Inc=TG_S['specimen_dec'][n],TG_S['specimen_inc'][n],
    TG_S_directions.append([Dec,Inc,1.])
TG_S_mean=pmag.fisher_mean(TG_S_directions)

#in TG08 and TG09, cut out clear outlier samples using distinct inclinations
TG_08 = TG_08[abs(TG_08['specimen_inc']) < 50]
TG_08 = TG_08[TG_08['specimen_inc'] < 0]
TG_08.reset_index(inplace=True, drop=True)
TG_09 = TG_09[abs(TG_09['specimen_inc']) < 50]
TG_09 = TG_09[TG_09['specimen_inc'] < 0]
TG_09.reset_index(inplace=True, drop=True)
TG_0809_tc_directions=[]
for n in range(len(TG_08)):
    Dec,Inc=TG_08['specimen_dec'][n],TG_08['specimen_inc'][n],
    TG_0809_tc_directions.append([Dec,Inc,1.])
for n in range(len(TG_09)):
    Dec,Inc=TG_09['specimen_dec'][n],TG_09['specimen_inc'][n],
    TG_0809_tc_directions.append([Dec,Inc,1.])

TG_0809_tc_mean=pmag.fisher_mean(TG_0809_tc_directions)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(TG_S_directions,color='black')
IPmag.iplotDI(TG_0809_tc_directions,color='red')
IPmag.iplotDImean(TG_S_mean['dec'],TG_S_mean['inc'],TG_S_mean["alpha95"],
                  color='yellow',marker='s',label='TG_S')
IPmag.iplotDImean(TG_0809_tc_mean['dec'],TG_0809_tc_mean['inc'],
                  TG_0809_tc_mean["alpha95"],color='b',marker='s',
                  label='TG_0809')
plt.show()
```



Results from the Timbavati region were split into two cooling units: Sites TG08 and TG09 are the northern intrusion and the rest of the sites belong to the southern intrusion.

In [46]:
```
#The following routine is meant to edit out some of the more anomalous...
#...data points using an a95 cutoff.
#As of now this new edited mean is not used in the data compilation
TG_S_directions_edited=[]
for n in range(len(TG_S_directions)):
    direction=(TG_S_directions[n][0],TG_S_directions[n][1])
    mean_direction=(TG_S_mean['dec'],TG_S_mean['inc'])
    threshold_angle=TG_S_mean['alpha95']*8 #this can be adjusted higher...
#...because the a95 is so small, but doesn't really change the mean direction
    if pmag.angle(direction,mean_direction) < threshold_angle:
        TG_S_directions_edited.append([TG_S_directions[n][0],
                                       TG_S_directions[n][1],1.])
#calculate and plot a new mean for TG-series with outliers removed
```

```
TG_S_edited_mean=pmag.fisher_mean(TG_S_directions_edited)
fignum = 1
plt.figure(num=fignum,figsize=(10,10))
IPmag.iplotNET(fignum)
IPmag.iplotDI(TG_S_directions_edited,color='black')
IPmag.iplotDImean(TG_S_edited_mean['dec'],TG_S_edited_mean['inc'],
                  TG_S_edited_mean["alpha95"],color='r',marker='s',
                  label='TG_S_edited_mean')
IPmag.iplotDImean(TG_S_mean['dec'],TG_S_mean['inc'],TG_S_mean["alpha95"],
                  color='b',marker='s',label='TG_S_mean')
IPmag.iplotDI(TG_0809_tc_directions,color='c')
IPmag.iplotDImean(TG_0809_tc_mean['dec'],TG_0809_tc_mean['inc'],
                  TG_0809_tc_mean["alpha95"],color='c',marker='s',
                  label='TG_0809_tc_mean')
```




```
In [47]: TG_S_edited_mean
```

```
Out[47]: {'alpha95': 4.0473231727096728,
          'csd': 20.541210839892674,
          'dec': 187.54542145044252,
          'inc': 2.8323609718223071,
          'k': 15.549554724702871,
          'n': 84,
          'r': 78.662226573720361}
```

```
In [48]: TG_S_mean

Out[48]: {'alpha95': 5.6822879739987684,
          'csd': 30.229395081132569,
          'dec': 186.26352347580931,
          'inc': 2.8748729267192652,
          'k': 7.1797797942023003,
          'n': 100,
          'r': 86.211276273411215}

In [49]: TG_0809_tc_mean

Out[49]: {'alpha95': 6.4825458482714726,
          'csd': 12.519051272281883,
          'dec': 182.80684734145208,
          'inc': -14.727279558982838,
          'k': 41.862696736281336,
          'n': 13,
          'r': 12.71334861498304}
```

2.4.5 Summary For All Seidel (2004) Cooling Units

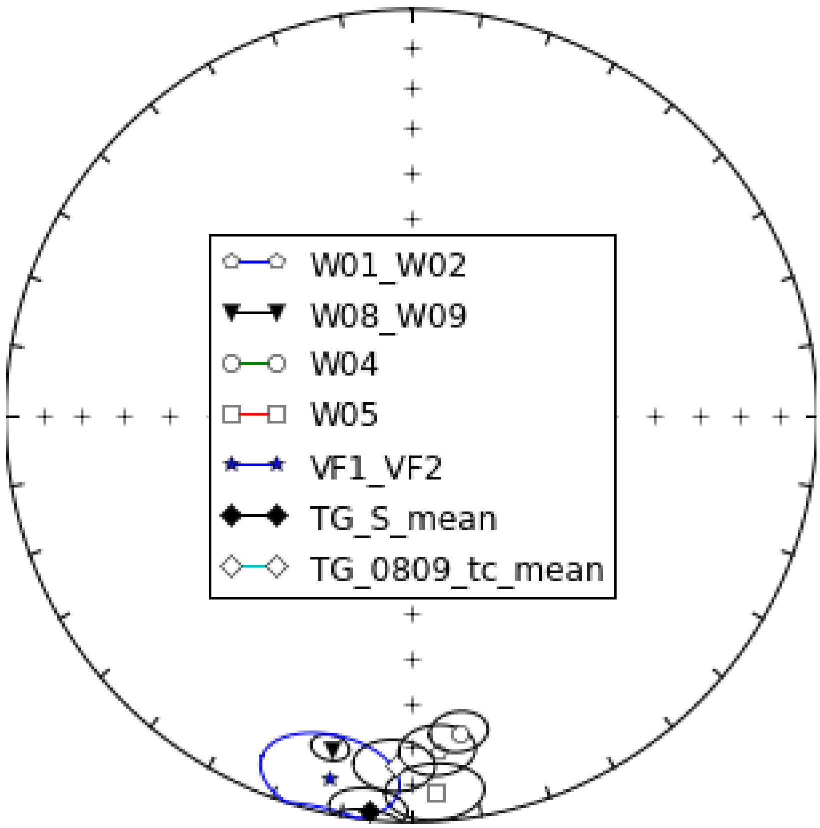
The TG-series has a number of localities, so the site latitude and longitude is given roughly for the center of the intrusion (median latitude and longitude of all Timbavati localities).

```
In [50]: cooling_unit_means.loc['TG-S-series'] = pd.Series({'site_ID':'TG-S-series',
                                                    'site_lat':-24.2,
                                                    'site_long':31.4,
                                                    'n':int(TG_tc_mean['n']),
                                                    'dec_tc':round(TG_S_mean['dec'],1),
                                                    'inc_tc':round(TG_S_mean['inc'],2),
                                                    'a_95':round(TG_S_mean['alpha95'],1),
                                                    'k':round(TG_S_mean['k'],1),
                                                    'date':1111.5,
                                                    'date_error':0.4})
         cooling_unit_means.loc['TG-N-series'] = pd.Series({'site_ID':'TG-N-series',
                                                    'site_lat':-23.2,
                                                    'site_long':31.2,
                                                    'n':int(TG_0809_tc_mean['n']),
                                                    'dec_tc':round(TG_0809_tc_mean['dec'],1),
                                                    'inc_tc':round(TG_0809_tc_mean['inc'],2),
                                                    'a_95':round(TG_0809_tc_mean['alpha95'],1),
                                                    'k':round(TG_0809_tc_mean['k'],1)})
         cooling_unit_means
```

Out[50]:

	site_ID	site_lat	site_long	n	dec_tc	inc_tc	a_95	k	date	date_error
W01_W02	W01_W02	-25.48	29.45	25	175.6	-18.40	6.2	22.6	NaN	NaN
W04	W04	-25.75	29.45	12	171.5	-22.30	4.9	80.8	NaN	NaN
W05	W05	-25.76	29.48	11	176.4	-7.80	7.5	38.0	NaN	NaN
W08_W09	W08_W09	-25.62	29.10	20	192.8	15.90	1.9	297.2	NaN	NaN
VF1_VF2	VF1_VF2	-25.80	27.50	21	7.2	-6.80	3.1	103.2	1108.6	1.2
TG-S-series	TG-S-series	-24.20	31.40	120	186.3	2.87	5.7	7.2	1111.5	0.4
TG-N-series	TG-N-series	-23.20	31.20	13	182.8	-14.73	6.5	41.9	NaN	NaN

In [51]:
```
fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(fignum)
IPmag.iplotDImean(W01_W02_tc_edited_mean['dec'],W01_W02_tc_edited_mean['inc'],
                W01_W02_tc_edited_mean["alpha95"],marker='p',label='W01_W02')
IPmag.iplotDImean(W08_W09_tc_mean['dec'],W08_W09_tc_mean['inc'],
                  W08_W09_tc_mean["alpha95"],marker='v',label='W08_W09')
IPmag.iplotDImean(W04_tc_mean['dec'],W04_tc_mean['inc'],
                  W04_tc_mean["alpha95"],marker='o',label='W04')
IPmag.iplotDImean(W05_tc_mean['dec'],W05_tc_mean['inc'],
                  W05_tc_mean["alpha95"],marker='s',label='W05')
#plotting opposite polarity for VF1/VF2 and VF4/VF5
IPmag.iplotDImean(VF1_VF2_tc_edited_mean['dec']+180,
                  VF1_VF2_tc_edited_mean['inc']*-1,
                  VF1_VF2_tc_edited_mean["alpha95"],marker='*',color='b',
                  label='VF1_VF2')
#VF4_VF5 not shown because we prefer data from Jones and McElhinny 1966 (J_M12)
IPmag.iplotDImean(TG_S_mean['dec'],TG_S_mean['inc'],TG_S_mean["alpha95"],
                  marker='D',label='TG_S_mean')
IPmag.iplotDImean(TG_0809_tc_mean['dec'],TG_0809_tc_mean['inc'],
                  TG_0809_tc_mean["alpha95"],marker='D',
                  label='TG_0809_tc_mean')
plt.legend(loc='center')
plt.show()
```



Sites that have reversed polarity are shown in blue (not including J_M12, which will be added later).

2.5 Data from Pancake (2001)

We evaluate data from the Pancake (2001) thesis in the same way we dealt with sites from the Seidel (2004) thesis. Maps are used to see which sites are sampled from the same intrusion, then we will see which of these sites have paleomagnetic results that could be combined. Sampling was conducted in southeast Botswana in two different regions.

```
In [52]: Pancake_Umk_ALL_SITES=Image(filename='Local_PNGs/Umk_Pancake01_ALL_Sites.png')
         display(Pancake_Umk_ALL_SITES)
```

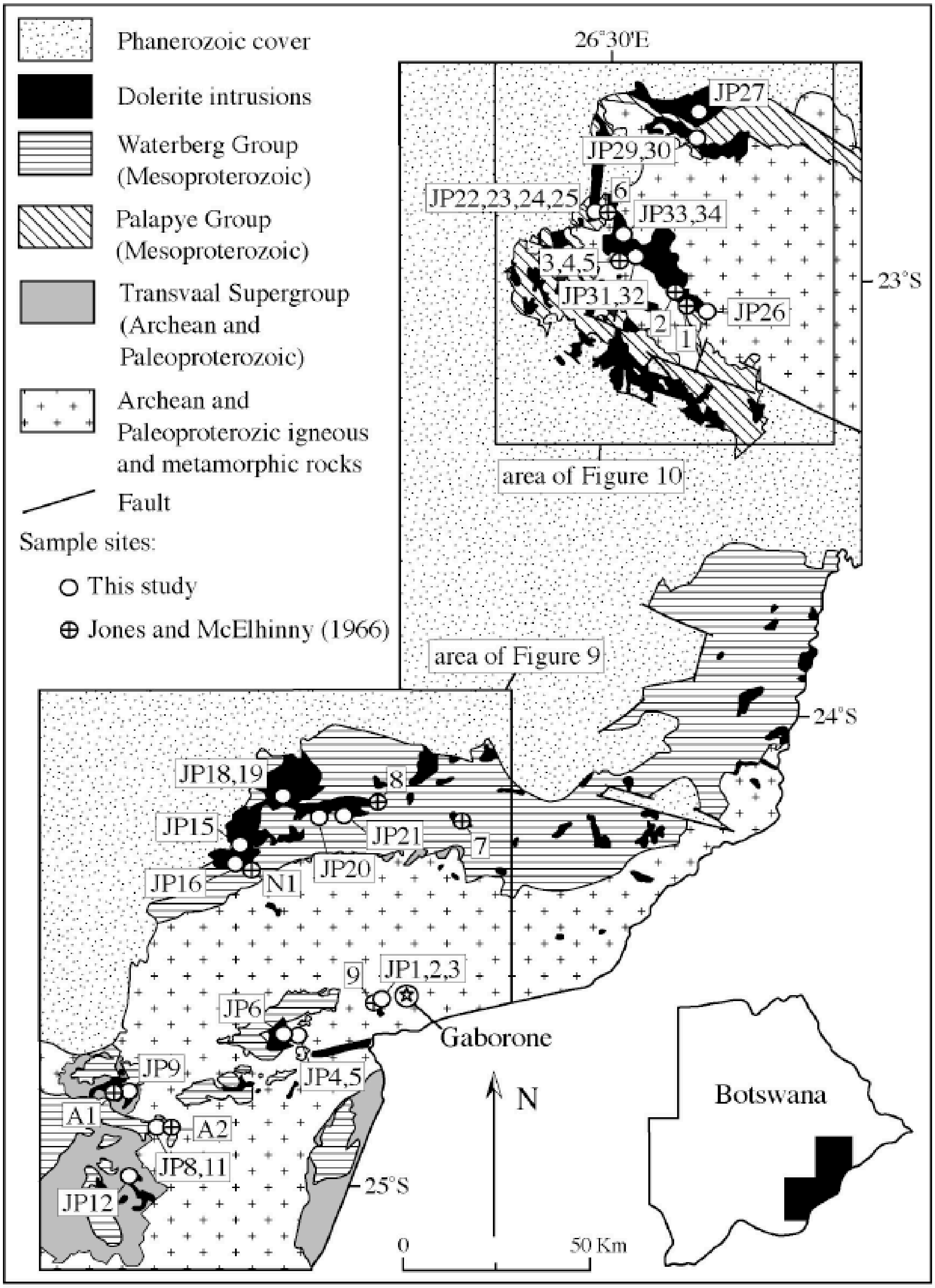


Figure 8. Geologic map of the study area showing location of paleomagnetic and geochronological sample sites. Areas of Figures 9 and 10 are indicated. Modified from Mortimer (1984).

More detailed maps are given for the two sub-regions:

```
In [53]: #images for first and second subplots
         Pancake_North = plt.imread('Local_PNGs/Umk_sites_Pancake01-North.png')
         Pancake_South = plt.imread('Local_PNGs/Umk_sites_Pancake01-South.png')
         #Make figure
         fig, (ax1, ax2) = plt.subplots(2,1, figsize=(50,30))
         ax1.imshow(Pancake_North,aspect=None)
         ax2.imshow(Pancake_South,aspect=None)
         ax1.set_axis_off() # Hide "spines"
         ax2.set_axis_off() # Hide "spines"
```

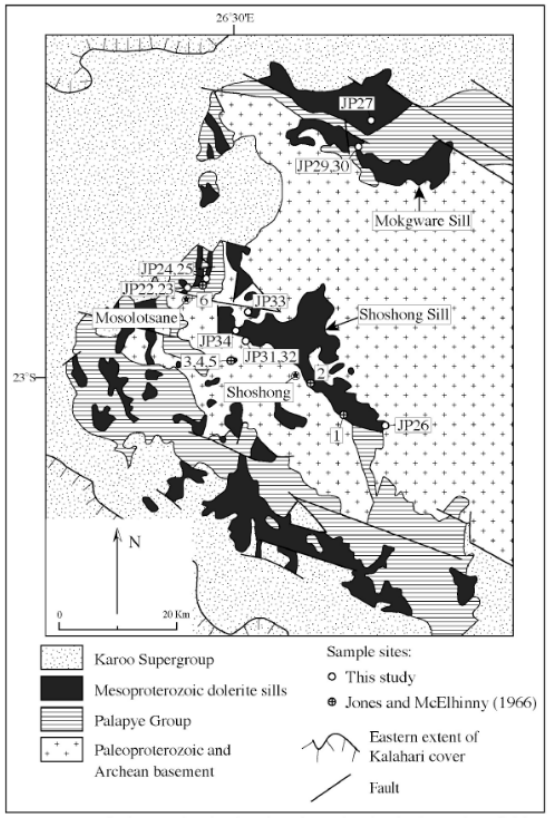
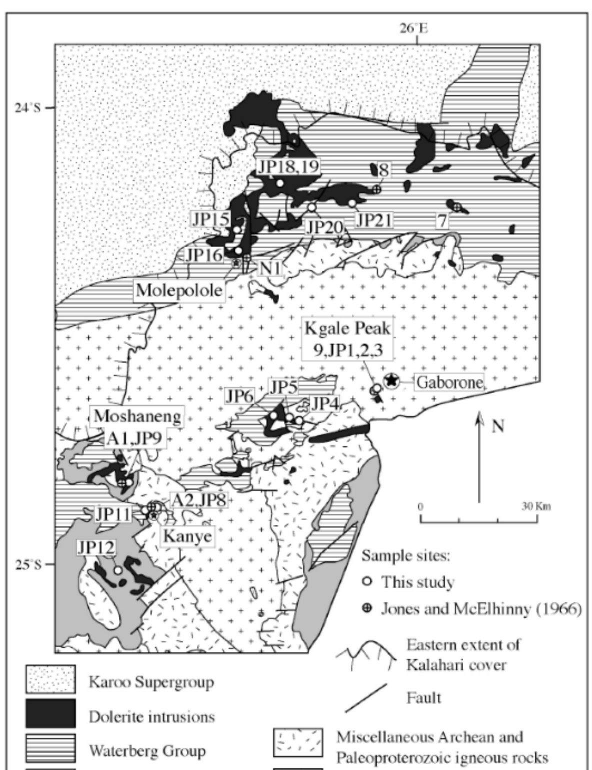


Figure 10. Geologic map showing location of samples sites in the northern field area. Modified from Mortimer (1984).



The raw demagnetization data from the Pancake (2001) thesis were digitized, converted to MagIC format, and then fit with vectors in order to combine sites from the same intrusions. Below is the sample magnetization data table for all of the sites/samples yielding stable data.

In [54]:
```
Pancake_dataframe=pd.read_csv('../Data/Prior_Data/Pancake_pmag_specimens.txt',
                               sep='\t',header=1)
Pancake_dataframe.head()
```

Out[54]:

	er_analyst_mail_names	er_citation_names	er_location_name	er_sample_name	er_site_name	er_specimen_name	magic_experiment_names	mag
0	NaN	This study	Pancake	JP1-1	JP1	JP1-1	JP1-1:LP-DIR-T:LP-DIR-AF	
1	NaN	This study	Pancake	JP1-1	JP1	JP1-1	JP1-1:LP-DIR-T:LP-DIR-AF	
2	NaN	This study	Pancake	JP1-1	JP1	JP1-1	JP1-1:LP-DIR-T:LP-DIR-AF	
3	NaN	This study	Pancake	JP1-2	JP1	JP1-2	JP1-2:LP-DIR-T:LP-DIR-AF	
4	NaN	This study	Pancake	JP1-2	JP1	JP1-2	JP1-2:LP-DIR-T:LP-DIR-AF	

We intend to use only line fits and tilt-corrected data, so we modify the table to only include those magnetizations.

In [55]:
```
#only include line fits
Pancake_DF = Pancake_dataframe[Pancake_dataframe[
        'specimen_direction_type'] == 'l']
#slice dataframe so that only tilt-corrected vectors are left
Pancake_DF_tc = Pancake_DF[Pancake_DF['specimen_tilt_correction'] == 100]
Pancake_DF_tc.reset_index(inplace=True)
Pancake_DF_tc.head()
```

Out[55]:

	index	er_analyst_mail_names	er_citation_names	er_location_name	er_sample_name	er_site_name	er_specimen_name	magic_experiment_names
0	5	NaN	This study	Pancake	JP1-2	JP1	JP1-2	JP1-2:LP-DIR-T:LP-DIR-A
1	8	NaN	This study	Pancake	JP1-3	JP1	JP1-3	JP1-3:LP-DIR-T:LP-DIR-A
2	11	NaN	This study	Pancake	JP1-4	JP1	JP1-4	JP1-4:LP-DIR-T:LP-DIR-A
3	14	NaN	This study	Pancake	JP1-5	JP1	JP1-5	JP1-5:LP-DIR-T:LP-DIR-A
4	17	NaN	This study	Pancake	JP1-6	JP1	JP1-6	JP1-6:LP-DIR-T:LP-DIR-A

2.5.1 Kgale Peak Intrusion

The first localities (numerically) are in the Kgale peak area, including a site (JM-9) from Jones and McElhinny (1966) shown on the map below. This sill was also sampled by Swanson-Hysell and Hanson as PW1 and PW2:

In [56]:
```
Pancake_Umk_Kgale=Image(filename=
                        'Local_PNGs/Umk_sites_Kgale_Peak_sill_Pancake01.png')
display(Pancake_Umk_Kgale)
```

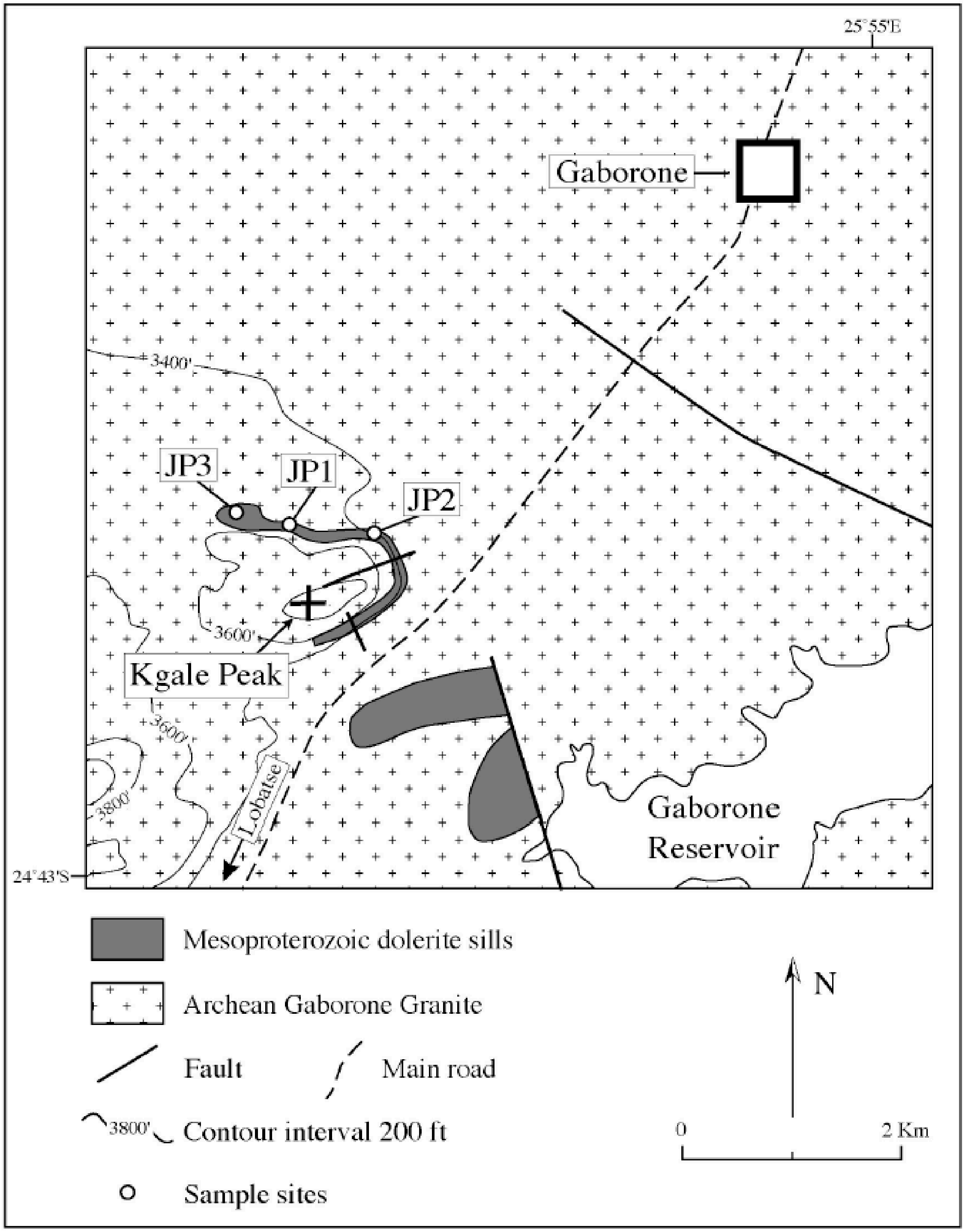


Figure 19. Geologic map of the area southwest of Gaborone showing the location of sample sites from the Kgale Peak sill. Modified from Key (1983).

Kgale Peak sill date A sample from JP1 was dated at 1108.0 0.9 Ma ($^{207}Pb/^{206}Pb$ weighted mean on baddeleyite; Hanson et al. 2004).

```
In [57]: JP1A_dates = [1108.2, 1109.0, 1107.1, 1108.2, 1107.7]
         JP1A_1sigma = [0.55, 2.4, 0.95, 1.95, 2.35]
         weighted_mean(JP1A_dates,JP1A_1sigma)
```

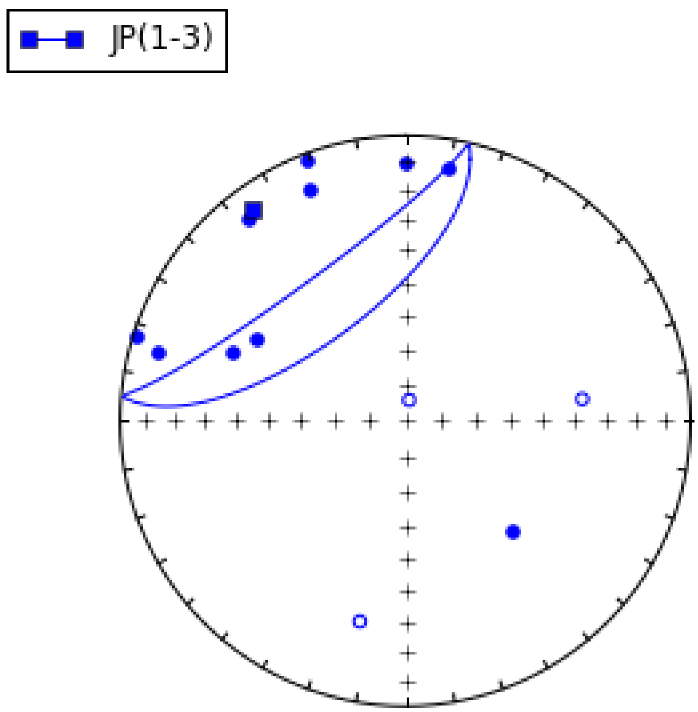
```
The weighted mean is:
1107.96736792
With a 2sigma error of:
0.891623717924
```

Kgale Peak sill paleomagnetism

```
In [58]: JP1_tc = Pancake_DF_tc.ix[Pancake_DF_tc['er_site_name'] == 'JP1']
         JP2_tc = Pancake_DF_tc.ix[Pancake_DF_tc['er_site_name'] == 'JP2']
         JP3_tc = Pancake_DF_tc.ix[Pancake_DF_tc['er_site_name'] == 'JP3']
         JP1_tc.reset_index(inplace=True)
         JP2_tc.reset_index(inplace=True)
         JP3_tc.reset_index(inplace=True)
```

```
In [59]: JP1_2_3_tc_directions=[]
         JP1_tc_directions=[]
         #create array of unit vectors from sample fits from sites Kgale sites
         for n in range(len(JP1_tc)):
             Dec,Inc=JP1_tc['specimen_dec'][n],JP1_tc['specimen_inc'][n]
             JP1_tc_directions.append([Dec,Inc,1.])
             JP1_2_3_tc_directions.append([Dec,Inc,1.])
         JP2_tc_directions=[]
         for n in range(len(JP2_tc)):
             Dec,Inc=JP2_tc['specimen_dec'][n],JP2_tc['specimen_inc'][n]
             JP2_tc_directions.append([Dec,Inc,1.])
             JP1_2_3_tc_directions.append([Dec,Inc,1.])
         JP3_tc_directions=[]
         for n in range(len(JP3_tc)):
             Dec,Inc=JP3_tc['specimen_dec'][n],JP3_tc['specimen_inc'][n]
             JP3_tc_directions.append([Dec,Inc,1.])
             JP1_2_3_tc_directions.append([Dec,Inc,1.])

         #save directions to be used in compilation mean because is same as PW1 and PW2
         f = open('../Data/Pickle/JP1_2_3_tc_directions.txt','w')
         pickle.dump(JP1_2_3_tc_directions, f)
```

```
JP1_2_3_tc_mean=pmag.fisher_mean(JP1_2_3_tc_directions)
fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(JP1_2_3_tc_directions,color='b')
IPmag.iplotDImean(JP1_2_3_tc_mean['dec'],JP1_2_3_tc_mean['inc'],
                  JP1_2_3_tc_mean["alpha95"],color='b',marker='s',
                  label='JP(1-3)')
```

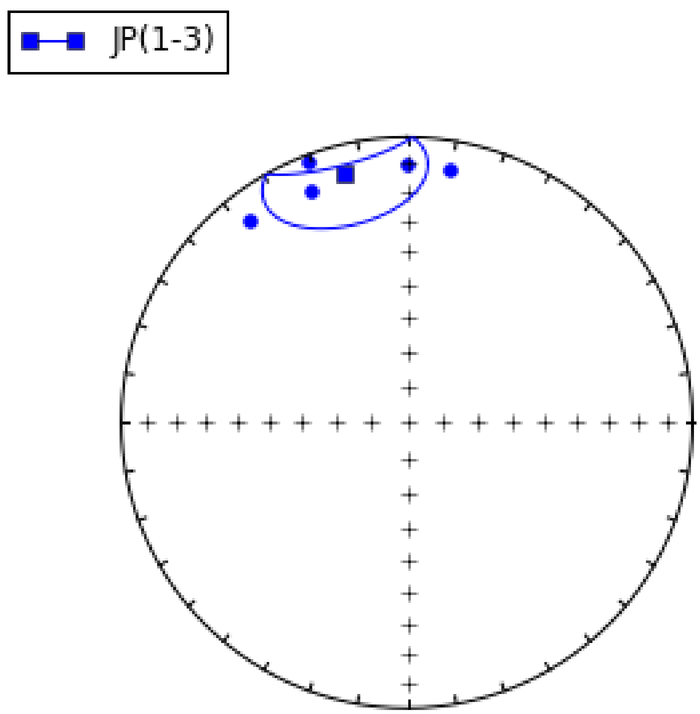


The directions from sites JP2 and JP3 account for most of the scatter, so we will only calculate a mean direction to JP1, which appears to be very different dec/inc given in Gose et al. (2006) for JP1 (Dec: 0.0, Inc: -0.1, a95: 21.6). Pancake (2001) describes types of magnetite grains in the rock, possibly associated with high temperature alteration:

> Hornblende and biotite occur as discrete grains of late-stage magmatic or high temperature deuteric origin. Magnetite occurs as discrete larger primary grains and as

> fine-grained or symplectic intergrowths with biotite, which are inferred to be of deuteric origin. Deuteric mineralization has resulted in formation of chlorite and epidote along joint surfaces and as alteration products of the primary phases (Eales, 1959). - (Pancake, 2001)

In [60]:
```
JP1_tc_mean=pmag.fisher_mean(JP1_tc_directions)
fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(JP1_tc_directions,color='b')
IPmag.iplotDImean(JP1_tc_mean['dec'],JP1_tc_mean['inc'],
                  JP1_tc_mean["alpha95"],color='b',marker='s',
                  label='JP(1-3)')
```



Jones and McElhinny (1966) determined this sill to have the opposite polarity (south-pointing declination). Our new data (sites PW1 and PW2) also found south-seeking magnetizations.

These results are puzzling when compared to the north-seeking results from JP1. Perhaps the results from JP(1-3) are not primary. Given the consistency between the Jones and McElhinny (1966) south-seeking polarity and the south-seeking polarity in the new data generated in this study for PW1 and PW2, we consider the south-seeking polarity to be primary and use data from our new sampling locales for the grand mean and polarity summary.

2.5.2 Manyana Sill (SW Kgale)

Three sites were sampled in the Manyana sill, which is just east of the eponymous town. A new site was sampled here in 2012 (NSH/RH), PW3. All of the sites studied by Pancake (2001) yielded very scattered demagnetization data, and some samples published in the Pancake thesis were not completely demagnetized. . . perhaps everything was finished for Gose et al. (2006)? Regardless, the data are not of use. The newly studied samples from the Manyana Sill, PW3, yielded data that can be used for this cooling unit. This new site was selected to try and avoid lightning struck areas.

```
In [61]: Pancake_Umk_Manyana=Image(filename=
                                   'Local_PNGs/Umk_sites_Manyana_Pancake01.png')
         display(Pancake_Umk_Manyana)
```

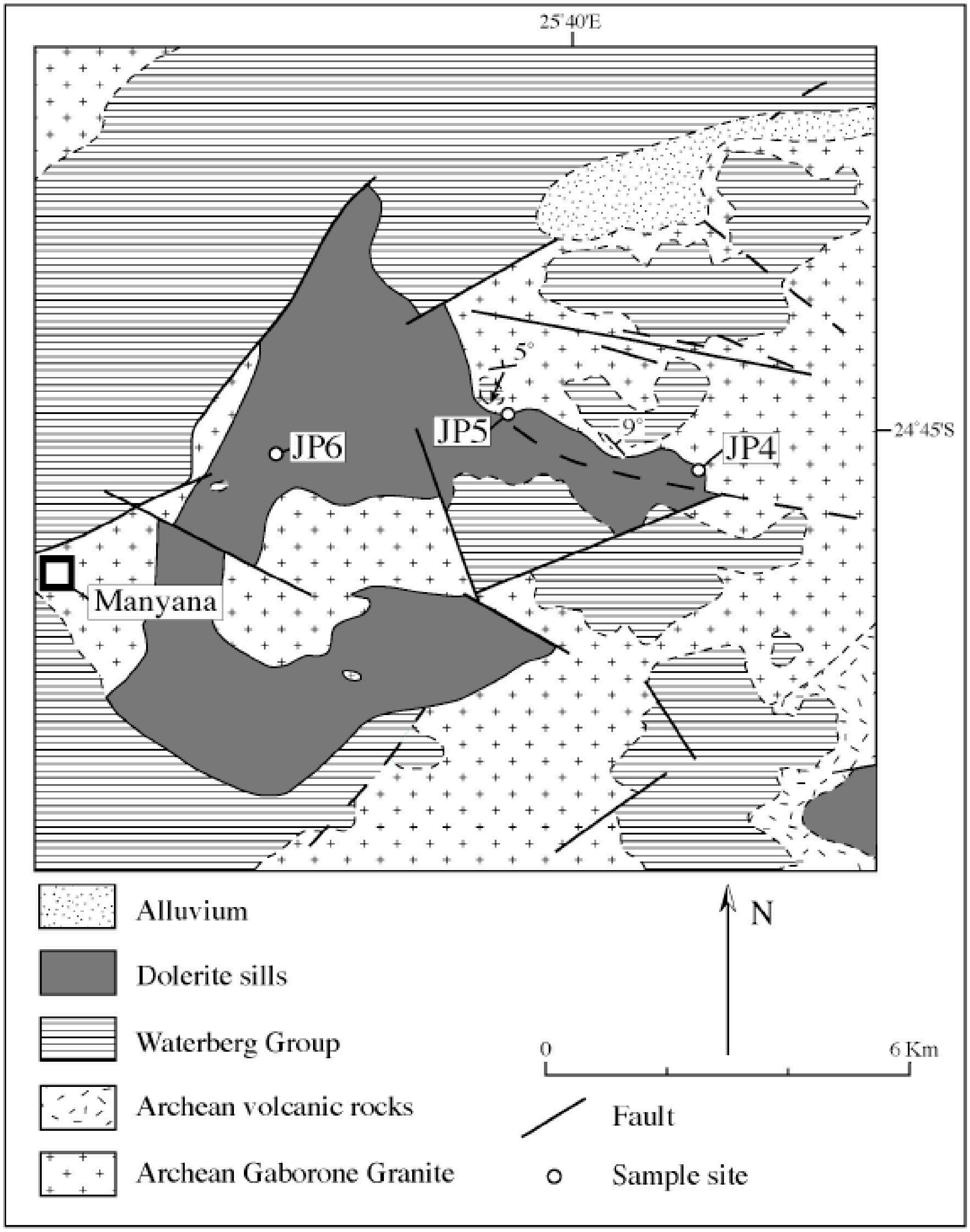


Figure 21. Geologic map showing location of sample sites JP4, 5, and 6. Modified from Key (1983).

> Pegmatitic dolerite occurs as discontinuous, subhorizontal layers, subvertical veins, and irregular pockets up to 40 cm across at this locality. Six cores were collected from a large exfoliation surface in typical dolerite at the very base of the outcrop, within the middle part of the sill. . . The granite here was partially melted during dolerite emplacement, implying that a significant volume of magma may have been transported laterally in the sill to provide the necessary heat for melting. (Pancake, 2001)

2.5.3 Moshaneng and Kanye Sills

One site was sampled in one sill near the town of Moshaneng, and two sites were sampled in the sill near the town Kanye. In addition, Jones and McElhinny (1966) sampled one site in each sill (J_M_A1 in Moshaneng and J_M_A2 in Kanye).

```
In [62]: Pancake_Umk_Moshaneng=Image(filename='Local_PNGs/Umk_sites_Moshaneng_sill_Pancake01.p
         display(Pancake_Umk_Moshaneng)
```

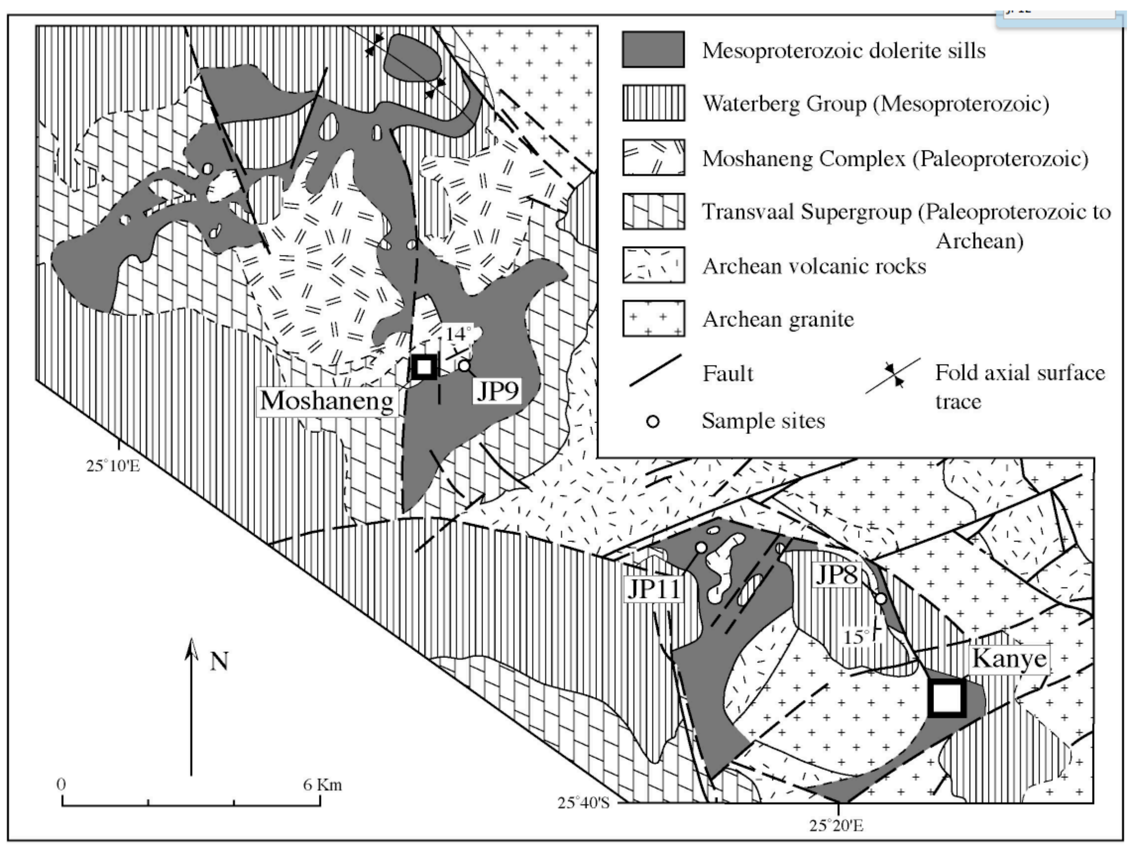


Figure 22. Geologic map of the area around Kanye showing the position of sample sites JP8, 9, and 11. Modified from Aldiss et al. (1989).

Site JP8 yields mostly scattered magnetization directions with only pairs of samples sharing similar directions. JP9 and JP11 each yield stable data. Only six samples are shown for site JP9 in the Pancake thesis. An additional eight samples from that site were collected by Hanson in 2002, after the Pancake thesis was completed, and were used to calculate the mean direction for the site given in Gose et al. (2006). The mean direction given by Jones and McElhinny (1966) for their site A1 is consistent with results from JP9 as expected given that the data are for the same intrusion.

The sill near Moshaneng has multiple magmatic ages at ca. 1927 Ma but also yields an K-Ar age of ˜1110 44 Ma (Gose et al., 2006). There is a concise discussion of these sites in Gose et al. (2006):

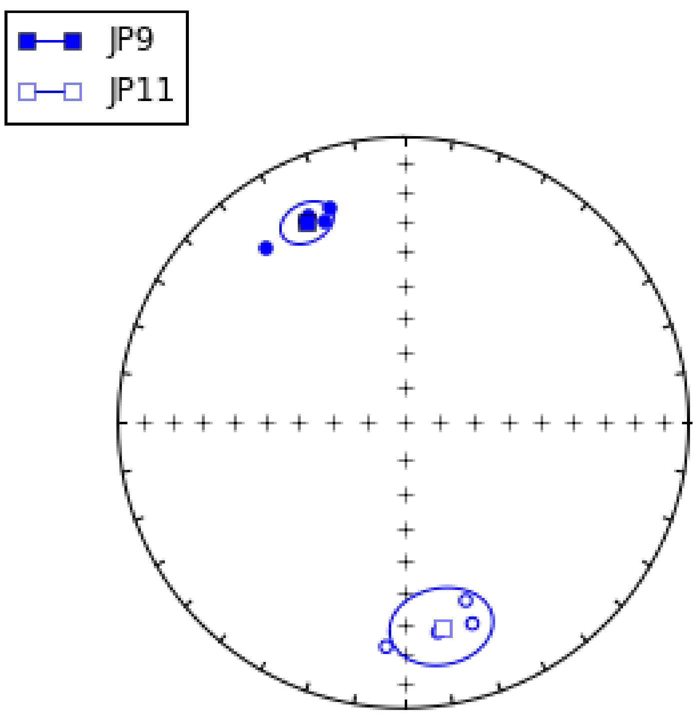
> . . . three separate samples from different parts of the Moshaneng Dolerite have yielded U-Pb baddeleyite ages of 1927.7 0.5 to 1927.1 0.7 Ma [Hanson et al., 2004b]. Two sites (JP9 and JP10) yielded magnetization directions markedly different from the directions for the Umkondo intrusions in the region (Figure 5) [Hanson et al., 2004b]. A third site in the Moshaneng Dolerite (JP11), which has yielded a U-Pb baddeleyite age of 1927.3 0.7 Ma, has a well-clustered magnetization direction statistically identical to directions for other dolerites in southeastern Botswana assigned to the Umkondo province (Table 1). It is likely that the dolerite at site JP11 was remagnetized during the Umkondo event, although no Umkondo intrusions are exposed in the vicinity. Jones and McElhinny [1966] reported a K-Ar date of 1110 44 Ma (recalculated to currently recommended constants) for a plagioclase-pyroxene fraction from a sample of the Moshaneng Dolerite collected near our sites JP9 and JP10, which suggests that these older dolerites were thermally disturbed during the Umkondo event.

Even though we discuss the results below, for the above reasons we exclude data from JP9 and JP11 from the Umkondo mean.

In [63]:
```
#Pull site/sample data from the table
JP9_tc = Pancake_DF_tc.ix[Pancake_DF_tc['er_site_name'] == 'JP9']
JP11_tc = Pancake_DF_tc.ix[Pancake_DF_tc['er_site_name'] == 'JP11']
JP9_tc.reset_index(inplace=True)
JP11_tc.reset_index(inplace=True)
#get rid of one random sample
JP11_tc = JP11_tc[JP11_tc['specimen_dec'] < 270]
```

In [64]:
```
JP9_tc_directions=[]
#create array of unit vectors from sample fits from sites Kanye sites
for n in range(len(JP9_tc)):
    Dec,Inc=JP9_tc['specimen_dec'][n],JP9_tc['specimen_inc'][n],
    JP9_tc_directions.append([Dec,Inc,1.])
JP11_tc_directions=[]
```

```
for n in range(len(JP11_tc)):
    Dec,Inc=JP11_tc['specimen_dec'][n],JP11_tc['specimen_inc'][n],
    JP11_tc_directions.append([Dec,Inc,1.])

JP9_tc_mean=pmag.fisher_mean(JP9_tc_directions)
JP11_tc_mean=pmag.fisher_mean(JP11_tc_directions)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(JP9_tc_directions,color='b')
IPmag.iplotDI(JP11_tc_directions,color='b')
IPmag.iplotDImean(JP9_tc_mean['dec'],JP9_tc_mean['inc'],
                  JP9_tc_mean["alpha95"],color='b',marker='s',label='JP9')
IPmag.iplotDImean(JP11_tc_mean['dec'],JP11_tc_mean['inc'],
                  JP11_tc_mean["alpha95"],color='b',marker='s',label='JP11')
```



> [At JP11] discontinuous zones of pegmatitic material with variable trends occur throughout the exposure. The sill has experienced deuteric alteration represented by replacement of pyroxene by hornblende, and formation of intergrowths of fine-grained magnetite and biotite... One thin pegmatitic vein was observed in the outcrop (Pancake, 2001)

> JP9 was collected from a large sill exposed in the Pioneer Quarry at Moshaneng... The dolerite at this site is fine grained and has intergranular texture (Fig. 14). The plagioclase is relatively unaltered but pyroxene shows partial uralitization (Pancake, 2001)

JP9 and JP11 each produce a VGP unless there is evidence that the sills are actually the same. This is unlikely because the directions are of opposite polarity. However, we have decided not to include results from the Moshaneng sill (JP9 and JP11) because U-Pb baddeleyite ages from the sill yielded an ages of ca. 1927 Ma (Hanson et al., 2004 *S. Afr. J. Geol.*).

In [65]:
```
#Add new data points to the unknown intrusion table
#...except none of these are added for reasons stated in the preceding text
unknown_intrusions.loc['JP9'] = pd.Series({'cooling_unit_name':'JP9',
                                           'site_lat':-24.96,
                                           'site_long':25.25,
                                           'n':int(JP1_tc_mean['n']),
                                           'dec_tc':round(JP9_tc_mean['dec'],1),
                                           'inc_tc':round(JP9_tc_mean['inc'],2),
                                           'a_95':round(JP9_tc_mean['alpha95'],1),
                                           'k':round(JP9_tc_mean['k'],1)})
unknown_intrusions.loc['JP11'] = pd.Series({'cooling_unit_name':'JP11',
                                           'site_lat':-24.94,
                                           'site_long':25.3,
                                           'n':int(JP11_tc_mean['n']),
                                           'dec_tc':round(JP11_tc_mean['dec'],1),
                                           'inc_tc':round(JP11_tc_mean['inc'],2),
                                           'a_95':round(JP11_tc_mean['alpha95'],1),
                                           'k':round(JP11_tc_mean['k'],1)})
```

2.5.4 JP12 - South of Kanye

There is one site sampled in a group of sills south of Kanye. The site yielded scattered results and will not be used in the compilation.

In [66]:
```
Pancake_Umk_KanyeS=Image(filename='Local_PNGs/Umk_sites_Kanye-S_JP-12_Pancake01.png')
display(Pancake_Umk_KanyeS)
```

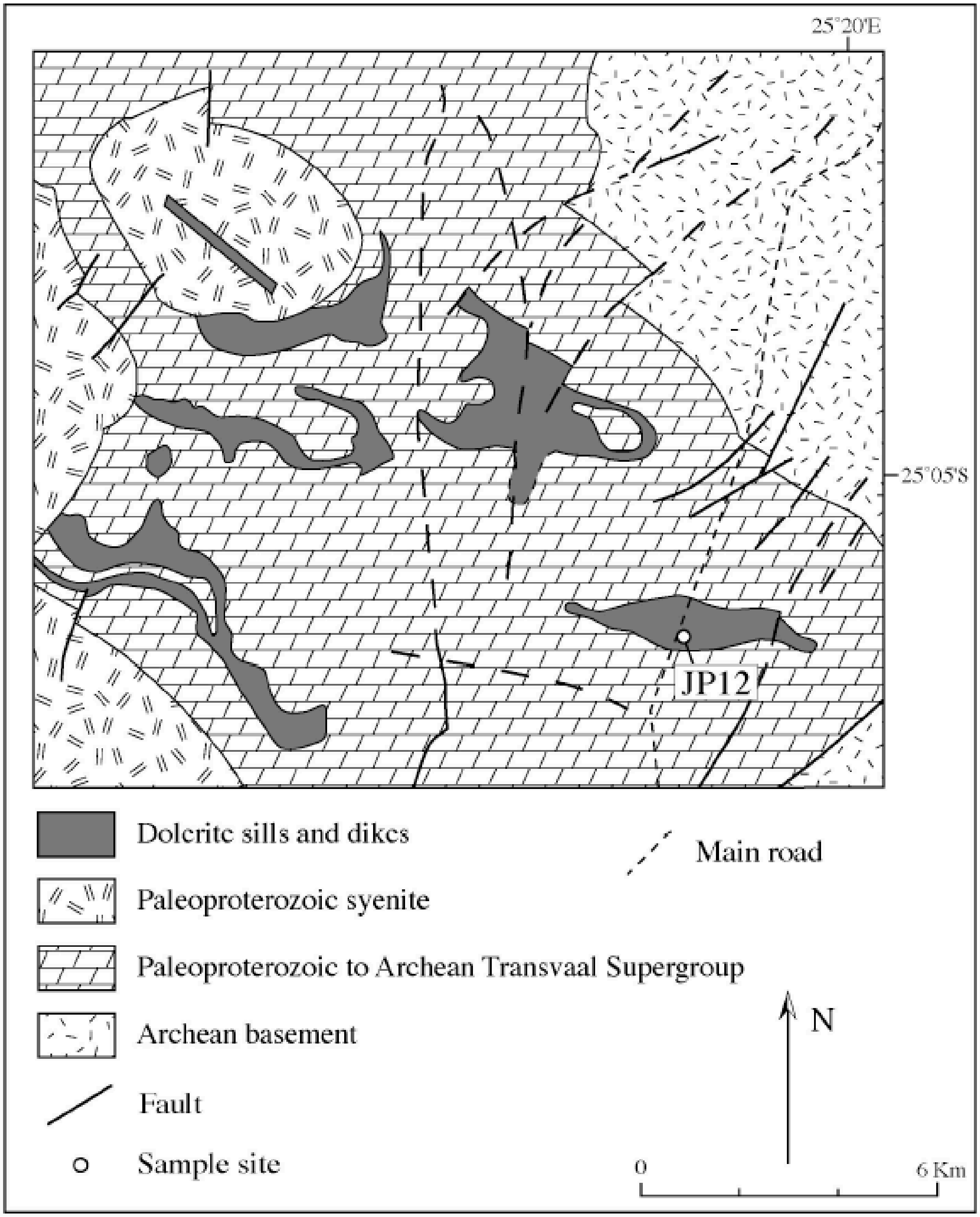


Figure 23. Geologic map of an area south of Kanye showing the position of sample site JP12. Modified from Mapeo (1994).

The sills are poorly exposed but appear to be concordant with the Transvaal strata.

> The age of these sills is unknown, and they could be significantly older than the Moshaneng dolerite. The dolerite at this site is distinctly different from other sills observed in the area because it contains abundant, large ophitic pyroxene crystals. (Pancake, 2001)

2.5.5 Molepolole Intrusions (JP(15, 16, 18, 19, 20, 21))

6 different localities were sampled in the area north of Molepolole by Pancake as can be seen in the map below. Jones and McElhinny (1966) also sampled at two localities in the area (site IDs: N1 and 8, found on regional map).

In [67]:
```
Pancake_Umk_Molepolole=Image(filename=
                                    'Local_PNGs/Umk_sites_Molepolole_Pancake01.png')
display(Pancake_Umk_Molepolole)
```

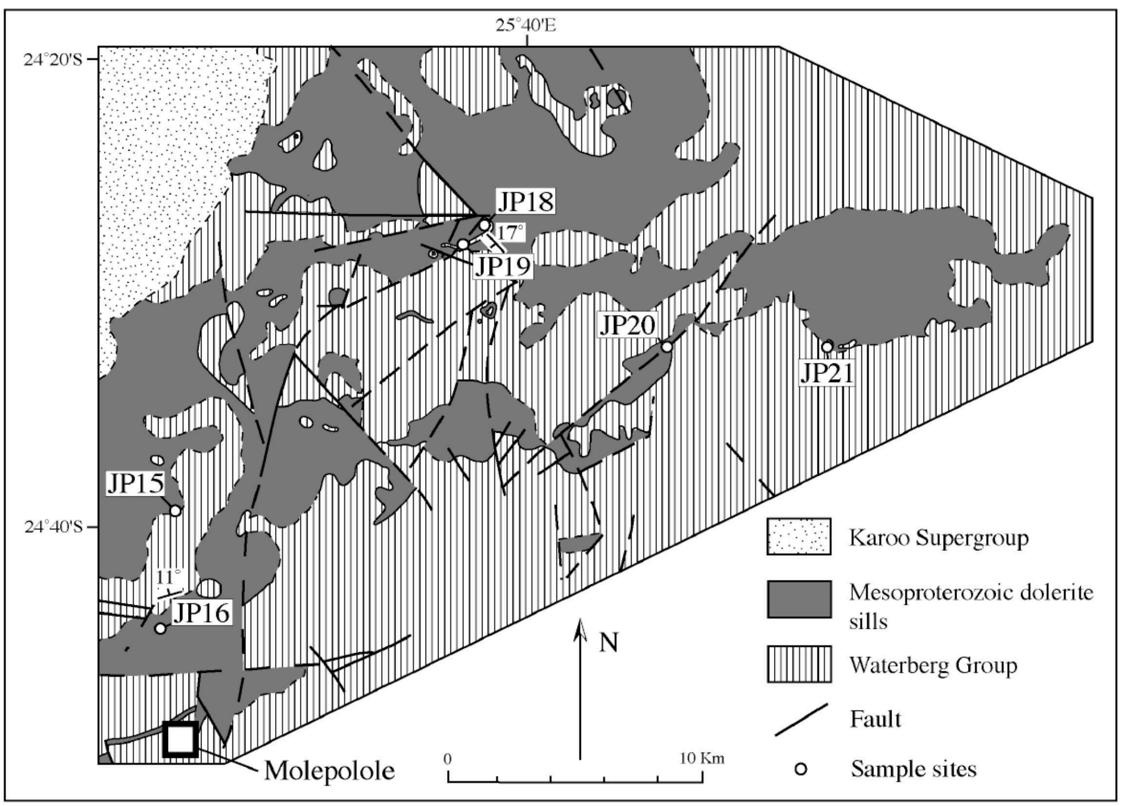


Figure 24. Geologic map of the area north and northeast of Molepolole showing the position of sample sites from this study. Modified from Jones (1973).

Some observations on the intrusion(s) as a whole: > The dolerite is shown on the geologic map as one large, continuous unit that intrudes Waterberg sedimentary units (Jones, 1973). The area is dissected by numerous faults. Field observations by the author indicate that a number of thick sills occur in this area but, because of generally poor exposure, it is impossible to determine the exact number of sills present or their dimensions. (Pancake, 2001)

Six sites were sampled by Pancake:

JP15: > Sample JP15 was collected from pavement exposure in an abandoned barrow pit. Field observations indicate that the site is located in the upper part of a sill, close to its contact with overlying Waterberg sandstone. The dolerite at this site is typically medium grained with common ophitic augite up to 1 cm across (Fig. 15). Plagioclase is generally altered to sericite and epidote, and pyroxene shows some uralitization. Ten cores were drilled in a small (< 3 m^2) weathered pavement exposure in the barrow pit. The site is unique in this study because the dolerite is not highly magnetized. (Pancake, 2001)

JP16: > The dolerite shows strong deuteric alteration. Plagioclase is largely replaced by epidote and sericite, and pyroxene is partly to completely replaced by chlorite and uralite. Ten cores were collected ~ 4 m from the upper margin of the sill, from four large, adjacent masses that appear to be in place but may have experienced some rotation during weathering and erosion. (Pancake, 2001)

JP18/JP19: > JP18 and 19 were collected from a single dolerite sill... The sill has experienced deuteric alteration, which is represented by replacement of plagioclase by sericite, and alteration of pyroxene to smectite and chlorite....[near JP19] has undergone significant deuteric alteration similar to that at site JP18 (Pancake, 2001)

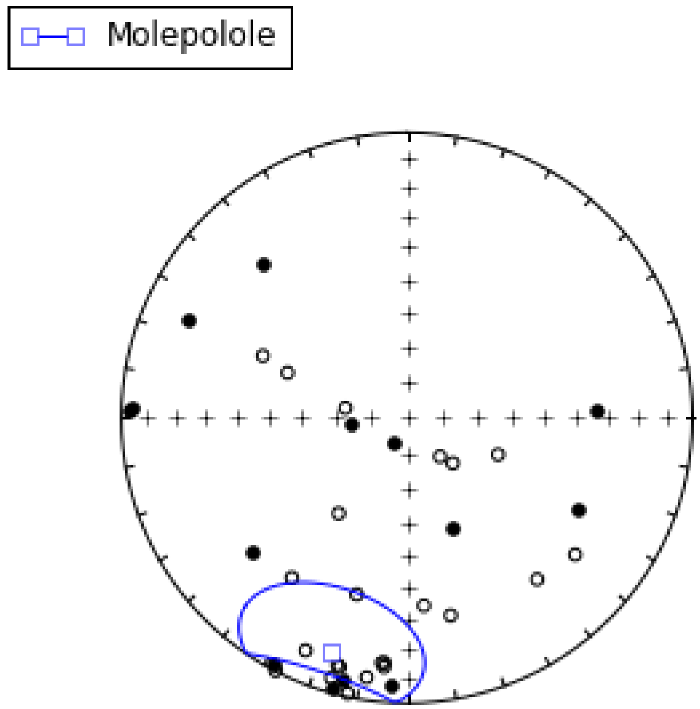
JP20: > The dolerite is uniformly fine grained and shows strong deuteric alteration. (Pancake, 2001)

JP21: > The dolerite here displays strong deuteric alteration... Cores 1-5 were collected from *in situ* pavement exposure 100 m south of cores 6-10. Cores 6-10 were drilled in several dolerite masses that could possibly have experienced some rotation.

Data from all sites are plotted below.

```
In [68]: Molep_tc = Pancake_DF_tc.ix[Pancake_DF_tc['er_location_name'] == 'Molepolole']
         Molep_tc.reset_index(inplace=True)
```

```
In [69]: Molep_tc_directions=[]
         for n in range(len(Molep_tc)):
             Dec,Inc=Molep_tc['specimen_dec'][n],Molep_tc['specimen_inc'][n],
             Molep_tc_directions.append([Dec,Inc,1.])
         Molep_tc_mean=pmag.fisher_mean(Molep_tc_directions)
         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(Molep_tc_directions,color='black')
         IPmag.iplotDImean(Molep_tc_mean['dec'],Molep_tc_mean['inc'],
                           Molep_tc_mean["alpha95"],color='b',marker='s',
                           label='Molepolole')
```



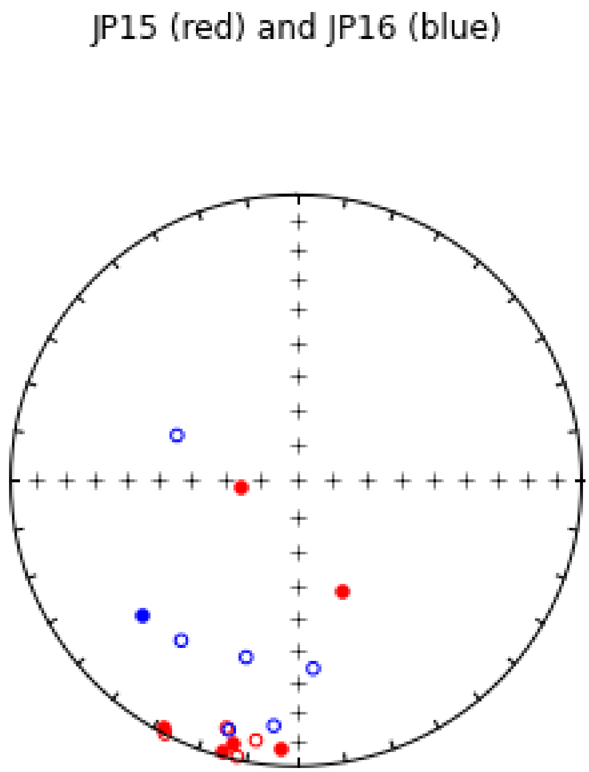
Only three of the sites shown have consistent/stable paleomagnetic results, therefore we will exclude the other three sites that have incoherent behavior (JP18, JP20, and JP21). This leaves JP15, JP16 and JP19 for consideration

Given their proximity and similar elevation, JP15 and JP16 appear to be from the same sill, which we refer to as the Suping Sill. Two of the new sites sampled by Swanson-Hysell and Hanson (PW11 and PW12) were quite close to JP15. At present, we consider JP19 to be from a distinct sill that we refer to as the Mabelwane Hill Sill given its distance from sites in the Suping Sill (15 km), the lack of continuity in outcrop expression and the distinct level it should represent based on the observed dip of the Suping Sill.

Suping Sill

```
In [70]: JP15 = Molep_tc.ix[Molep_tc['er_site_name'] == 'JP15']
         JP16 = Molep_tc.ix[Molep_tc['er_site_name'] == 'JP16']
         JP16 = JP16.drop('level_0',1)
```

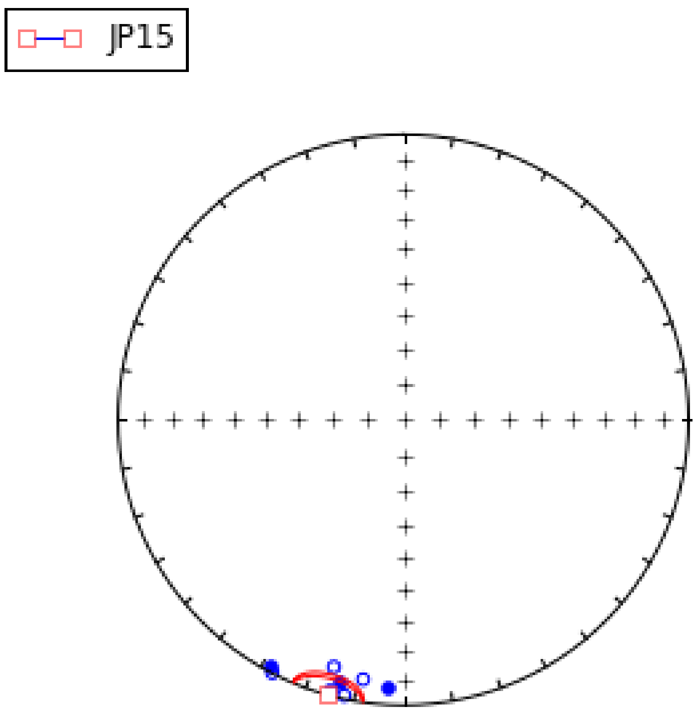
```
JP16.reset_index(inplace=True)

JP15_tc_directions=[]
for n in range(len(JP15)):
    Dec,Inc=JP15['specimen_dec'][n],JP15['specimen_inc'][n]
    JP15_tc_directions.append([Dec,Inc,1.])

JP15_tc_mean = pmag.fisher_mean(JP15_tc_directions)

JP16_tc_directions=[]
for n in range(len(JP16)):
    Dec,Inc=JP16['specimen_dec'][n],JP16['specimen_inc'][n]
    JP16_tc_directions.append([Dec,Inc,1.])

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(fignum)
IPmag.iplotDI(JP15_tc_directions,color='r')
IPmag.iplotDI(JP16_tc_directions,color='b')
plt.title('JP15 (red) and JP16 (blue)')
plt.show()
```



JP15 has a coherent low inclination population with two outliers. These two outliers are removed below and the edited data will be combined with the new results from PW11 and PW12.

```
In [71]: JP15_tc_dir_edited=[]
         for n in range(len(JP15_tc_directions)):
             direction=(JP15_tc_directions[n][0],JP15_tc_directions[n][1])
             mean_direction=(JP15_tc_mean['dec'],JP15_tc_mean['inc'])
             threshold_angle=JP15_tc_mean['alpha95']*2
             if pmag.angle(direction,mean_direction) < threshold_angle:
                 JP15_tc_dir_edited.append([JP15_tc_directions[n][0],
                                            JP15_tc_directions[n][1],1.])
         #calculate and plot a new mean for with outliers removed
         JP15_tc_edited_mean=pmag.fisher_mean(JP15_tc_dir_edited)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
```

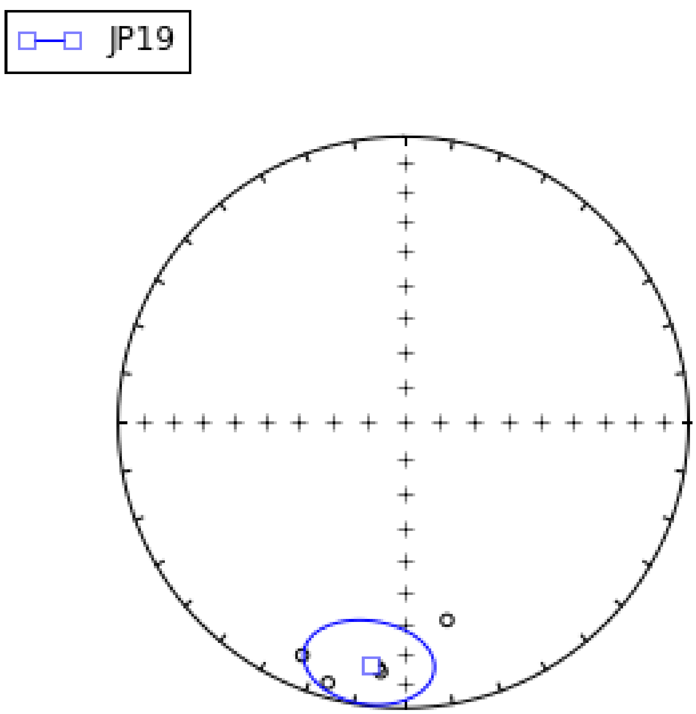
```
IPmag.iplotNET(fignum)
IPmag.iplotDI(JP15_tc_dir_edited, color='b')
IPmag.iplotDImean(JP15_tc_edited_mean['dec'],JP15_tc_edited_mean['inc'],
                  JP15_tc_edited_mean["alpha95"],color='r',marker='s',
                  label='JP15')
```



Mabelwane Hill sill

```
In [72]: JP19 = Molep_tc.ix[Molep_tc['er_site_name'] == 'JP19']
         JP19 = JP19.drop('level_0',1)
         JP19.reset_index(inplace=True)

         JP19_tc_directions=[]
         for n in range(len(JP19)):
```

```
    Dec,Inc=JP19['specimen_dec'][n],JP19['specimen_inc'][n],
    JP19_tc_directions.append([Dec,Inc,1.])
JP19_mean=pmag.fisher_mean(JP19_tc_directions)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(fignum)
IPmag.iplotDI(JP19_tc_directions,color='k')
IPmag.iplotDImean(JP19_mean['dec'],JP19_mean['inc'],JP19_mean["alpha95"],
                  color='b',marker='s',label='JP19')
```




```
In [73]: #Add new data points to cooling unit table
         cooling_unit_means.loc['JP15'] = pd.Series({'site_ID':'JP15',
                                          'site_lat':-24.32,
                                          'site_long':25.53,
```

```
                                        'n':int(JP15_tc_edited_mean['n']),
                                        'dec_tc':round(JP15_tc_edited_mean['dec'],1),
                                        'inc_tc':round(JP15_tc_edited_mean['inc'],2),
                                        'a_95':round(JP15_tc_edited_mean['alpha95'],1),
                                        'k':round(JP15_tc_edited_mean['k'],1)})

cooling_unit_means.loc['JP19'] = pd.Series({'site_ID':'JP19',
                                        'site_lat':-24.23,
                                        'site_long':25.64,
                                        'n':int(JP19_mean['n']),
                                        'dec_tc':round(JP19_mean['dec'],1),
                                        'inc_tc':round(JP19_mean['inc'],2),
                                        'a_95':round(JP19_mean['alpha95'],1),
                                        'k':round(JP19_mean['k'],1)})
```

2.5.6 Mosolotsane 1 Intrusion

Near the town of Mosolotsane, Pancake (2001) sampled for paleomagnetism from three localities: JP22, JP23 and JP24 (see map above).

> Samples JP22-24... are inferred to come from a single north-trending, gently west-dipping sill, based on topographic relations between the separate exposures. The sill... is unusually felsic rich.... **In general the samples are not highly altered**. (Pancake, 2001)

In [74]:
```
Pancake_Umk_MosoShosh=Image(filename=
                    'Local_PNGs/Umk_Sites_Mosolotsane-Shoshong_Pancake01.png')
display(Pancake_Umk_MosoShosh)
```

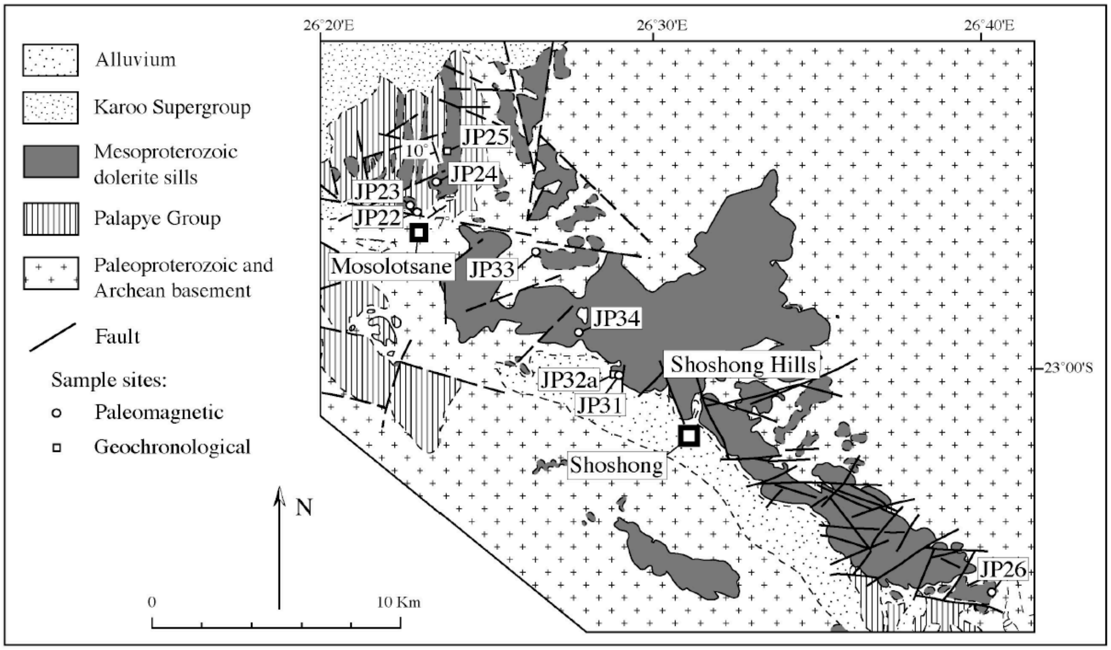


Figure 25. Geologic map of the area around the Shoshong Hills showing the position of sample sites from this study. Modified from Skinner (1978a, b) and Ermanovics (1978, 1980).

Mosolotsane 1 date Two samples were collected for geochronology: JP24 and JP25 and 207Pb/206Pb weighted mean dates on baddeleyite were published in Hanson et al. (2004):

- JP24: 1109.3 0.8 Ma
- JP25: 1109.3 1.0 Ma

These two geochronologic samples are from the same sill (note the similarity of the dates) and so a single weighted mean should be calculated from the individual baddeyite grain dates published in Hanson et al. (2004). This recalculation is done in the code below:

```
In [75]: JP24_dates = [1109.5, 1108.9, 1109.6, 1109.0]
         JP24_1sigma = [.75, 1.0, 0.9, 0.8]

         print 'JP24'
         weighted_mean(JP24_dates,JP24_1sigma)

         JP25_dates = [1108.4, 1109.7, 1109.3, 1110.1]
         JP25_1sigma = [.95, 0.95, 0.8, 1.5]
         print ' '
         print 'JP25'
```

```
weighted_mean(JP25_dates,JP25_1sigma)

JP24_JP25_dates = JP24_dates + JP25_dates
JP24_JP25_1sigma = JP24_1sigma + JP25_1sigma
print ' '
print 'JP24 and JP25 combined'
weighted_mean(JP24_JP25_dates,JP24_JP25_1sigma)
```

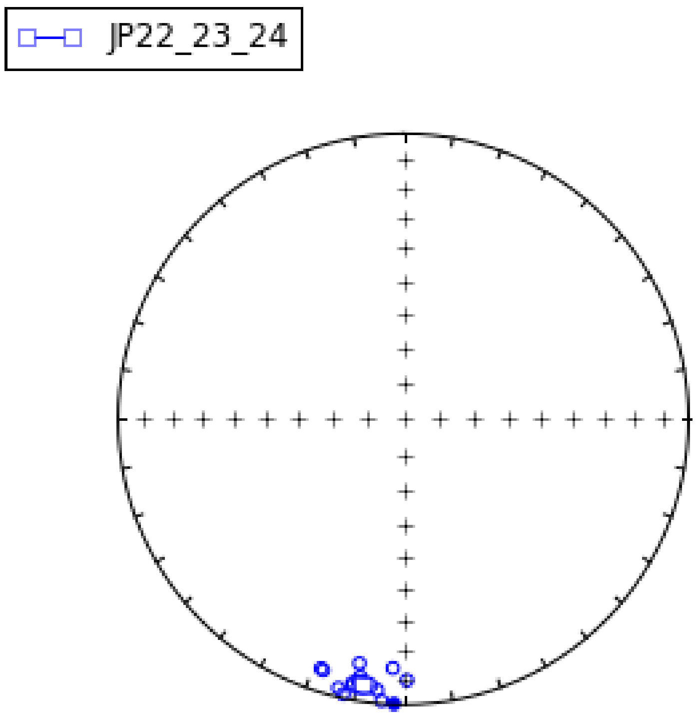
```
JP24
The weighted mean is:
1109.27438062
With a 2sigma error of:
0.847058823529

JP25
The weighted mean is:
1109.25300484
With a 2sigma error of:
0.973237631429

JP24 and JP25 combined
The weighted mean is:
1109.26516737
With a 2sigma error of:
0.638946442001
```

Mosolotsane 1 paleomagnetism

```
In [76]: JP22= Pancake_DF_tc.ix[Pancake_DF_tc['er_site_name'] == 'JP22']
         JP22.reset_index(inplace=True)
         JP23 = Pancake_DF_tc.ix[Pancake_DF_tc['er_site_name'] == 'JP23']
         JP23.reset_index(inplace=True)
         JP24 = Pancake_DF_tc.ix[Pancake_DF_tc['er_site_name'] == 'JP24']
         JP24.reset_index(inplace=True)
```

```
In [77]: JP22_23_24_tc_directions=[]

         for n in range(len(JP22)):
             Dec,Inc=JP22['specimen_dec'][n],JP22['specimen_inc'][n]
             JP22_23_24_tc_directions.append([Dec,Inc,1.])
         for n in range(len(JP23)):
             Dec,Inc=JP23['specimen_dec'][n],JP23['specimen_inc'][n]
             JP22_23_24_tc_directions.append([Dec,Inc,1.])
         for n in range(len(JP24)):
```

```
    Dec,Inc=JP24['specimen_dec'][n],JP24['specimen_inc'][n]
    JP22_23_24_tc_directions.append([Dec,Inc,1.])
JP22_23_24_tc_mean=pmag.fisher_mean(JP22_23_24_tc_directions)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(fignum)
IPmag.iplotDI(JP22_23_24_tc_directions,color='b')
IPmag.iplotDImean(JP22_23_24_tc_mean['dec'],JP22_23_24_tc_mean['inc'],
                  JP22_23_24_tc_mean["alpha95"],color='b',marker='s',
                  label='JP22_23_24')
```




```
In [78]: #Add new data points to cooling unit table
         cooling_unit_means.loc['Mosolotsane 1'] = pd.Series({'site_ID':
                                          'JP22, JP23, JP24',
```

```
'site_lat':-22.91,
'site_long':26.39,
'n':int(JP22_23_24_tc_mean['n']),
'dec_tc':round(JP22_23_24_tc_mean['dec'],1),
'inc_tc':round(JP22_23_24_tc_mean['inc'],2),
'a_95':round(JP22_23_24_tc_mean['alpha95'],1),
'k':round(JP22_23_24_tc_mean['k'],1),
'date':1109.3,
'date_error':0.6})
```

2.5.7 Shoshong Sill

Sites JP26, JP31, JP33 and JP34 were collected from the thick Shoshong Sill. This sill was also sampled by Jones and McElhinny (1966) as sites 1, 2, 3, 4, 5, 6 and by Swanson-Hysell and Hanson as site PW28. A description is provided by Pancake (2001):

> JP26 and 31-34 were collected from the areally extensive, thick (>120 m in places) dolerite sill that caps the Shoshong Hills. . . According to Grubb (1961), several vertical petrographic zones occur in the sill. The lower two zones, which make up over half of the sill, are bronzite-phyric dolerite. These two zones are separated by an interval of rhythmic layering defined by orthopyroxene-rich layers several centimeters thick (Fig. 13). The third zone lacks orthopyroxene and contains weakly subophitic augite and pigeonite. The upper 11 m of the sill is characterized by abundant pegmatitic segregations.

Shoshong dates Samples JP32a and JP35 were both collected from the large Shosong sill. Information about JP32a is provided in Pancake (2001):

> Sample JP32a was collected from the southern part of the large Shoshong sill northwest of the village of Shoshong (Fig. 25). This sample is ˜ 100 m west of paleomagnetic sample site JP31 and is separated from that site by a small, northeast-trending valley. The rock at site JP32a is significantly more felsic than that at the paleomagnetic sample site. It is inferred to occupy a down-faulted block derived from the felsic, pegmatite-rich, upper part of the sill described by Grubb (1961). The fault separating the two sites appears to lie within the valley running between them. This fault is not shown by Skinner (1978a), although he does show another fault subparallel to it and a short distance east of site JP31 (Fig. 25). The felsic-rich rock at site JP32a is medium-grained, clinopyroxene-rich granite with abundant interstitial granophyre and is typical of the more differentiated portions of the sills in the northern part of the study area. Coarser grained, discontinuous pegmatitic segregations several centimeters across show gradational contacts with the host rock. The geochronological sample comes from part of the granite that contains abundant pegmatitic segregations.

Dates for samples JP32A and JP35 were presented separately in Hanson et al., (2004) with very similar weighted mean dates. In the code below, we calculate a combined weighted mean for the samples combined given that each of the geochronology samples come from the same cooling unit.

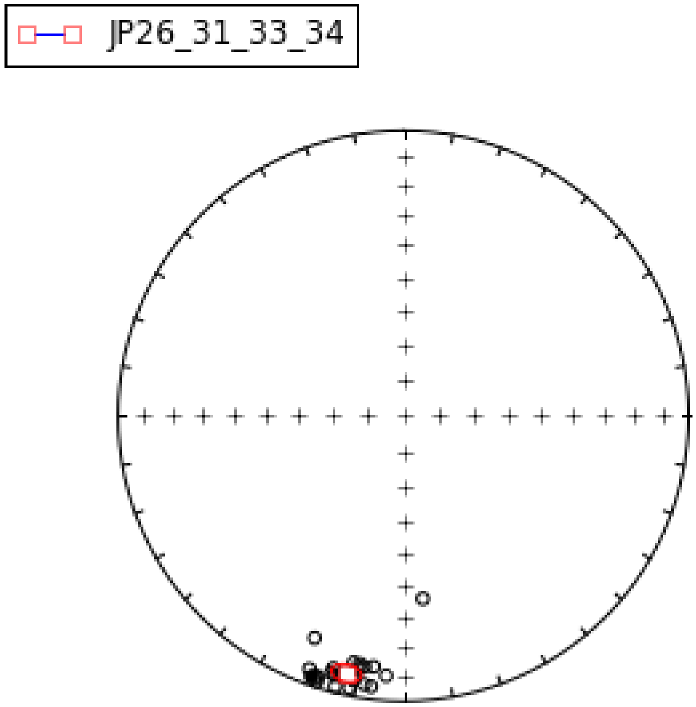
```
In [79]: JP32A_dates = [1109.2, 1109.9, 1109.1, 1110.1,
                        1108.5, 1109.1, 1108.5, 1109.0, 1109.0]
         JP32A_1sigma = [0.45, 0.8, 0.6, 0.45, 0.8, 0.6, 1.1, 0.6, 0.8]
         print 'JP32A'
         weighted_mean(JP32A_dates,JP32A_1sigma)

         JP35_dates = [1110.4, 1108.7, 1108.6, 1107.3, 1109.2]
         JP35_1sigma = [0.55, 0.9, 1.2, 1.55, 0.6]
         print ' '
         print 'JP35'
         weighted_mean(JP35_dates,JP35_1sigma)

         JP32A_JP35_dates = JP32A_dates + JP35_dates
         JP32A_JP35_1sigma = JP32A_1sigma + JP35_1sigma

         print ' '
         print 'JP32A and JP35 combined'
         weighted_mean(JP32A_JP35_dates,JP32A_JP35_1sigma)
```

```
JP32A
The weighted mean is:
1109.30294803
With a 2sigma error of:
0.410617693992

JP35
The weighted mean is:
1109.45414609
With a 2sigma error of:
0.688885282338

JP32A and JP35 combined
The weighted mean is:
1109.34258459
With a 2sigma error of:
0.352713218928
```

Shoshong paleomagnetism

```
In [80]: JP26 = Pancake_DF_tc.ix[Pancake_DF_tc['er_site_name'] == 'JP26']
         JP26.reset_index(inplace=True)
```

```
JP31 = Pancake_DF_tc.ix[Pancake_DF_tc['er_site_name'] == 'JP31']
JP31.reset_index(inplace=True)
JP33 = Pancake_DF_tc.ix[Pancake_DF_tc['er_site_name'] == 'JP33']
JP33.reset_index(inplace=True)
JP34 = Pancake_DF_tc.ix[Pancake_DF_tc['er_site_name'] == 'JP34']
JP34.reset_index(inplace=True)

JP26_31_33_34_tc_directions=[]
for n in range(len(JP26)):
    Dec,Inc=JP26['specimen_dec'][n],JP26['specimen_inc'][n]
    JP26_31_33_34_tc_directions.append([Dec,Inc,1.])
for n in range(len(JP31)):
    Dec,Inc=JP31['specimen_dec'][n],JP31['specimen_inc'][n]
    JP26_31_33_34_tc_directions.append([Dec,Inc,1.])
for n in range(len(JP33)):
    Dec,Inc=JP33['specimen_dec'][n],JP33['specimen_inc'][n]
    JP26_31_33_34_tc_directions.append([Dec,Inc,1.])
for n in range(len(JP34)):
    Dec,Inc=JP34['specimen_dec'][n],JP34['specimen_inc'][n]
    JP26_31_33_34_tc_directions.append([Dec,Inc,1.])
JP26_31_33_34_tc_mean=pmag.fisher_mean(JP26_31_33_34_tc_directions)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(fignum)
IPmag.iplotDI(JP26_31_33_34_tc_directions,color='k')
IPmag.iplotDImean(JP26_31_33_34_tc_mean['dec'],JP26_31_33_34_tc_mean['inc'],
                  JP26_31_33_34_tc_mean["alpha95"],color='r',marker='s',
                  label='JP26_31_33_34')
```




```
In [81]: cooling_unit_means.loc['Shoshong'] = pd.Series({'site_ID':
                                         'JP26, JP31, JP33, JP34)',
                                         'site_lat':-23.00,
                                         'site_long':26.48,
                                         'n':int(JP26_31_33_34_tc_mean['n']),
                                         'dec_tc':round(JP26_31_33_34_tc_mean['dec'],1),
                                         'inc_tc':round(JP26_31_33_34_tc_mean['inc'],2),
                                         'a_95':round(JP26_31_33_34_tc_mean['alpha95'],1),
                                         'k':round(JP26_31_33_34_tc_mean['k'],1),
                                         'date':1109.3,
                                         'date_error':0.4})
```

2.5.8 Mokgware intrusions

Three sites were sampled in this area with only one yielding stable and consistent magnetic behavior, JP30.

```
In [82]: Pancake_Umk_Mokgware=Image(filename=
                              'Local_PNGs/Umk_sites_Mokgware_Pancake01.png')
         display(Pancake_Umk_Mokgware)
```

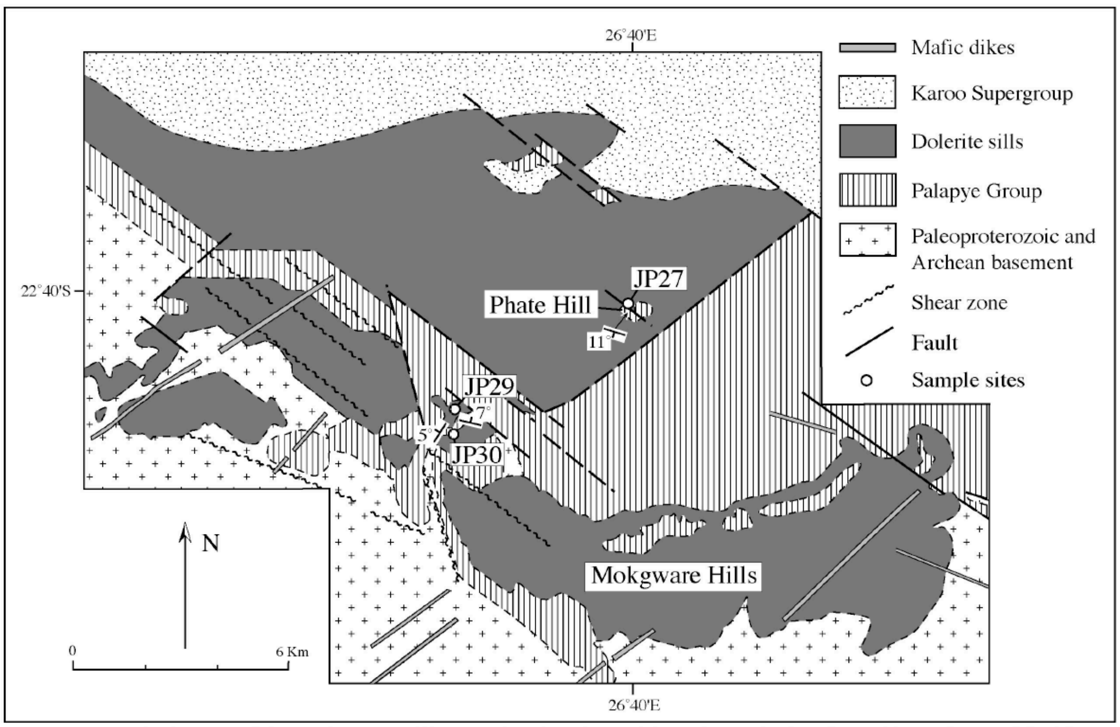


Figure 29. Geologic map of the Mokgware Hills area showing the position of sample sites from this study. Modified from Ermanovics (1980).

Mokgware Sill date A sample from JP1 was dated at 1112.0 0.5 Ma (^{207}Pb/^{206}Pb weighted mean on baddeleyite; Hanson et al. 2004).

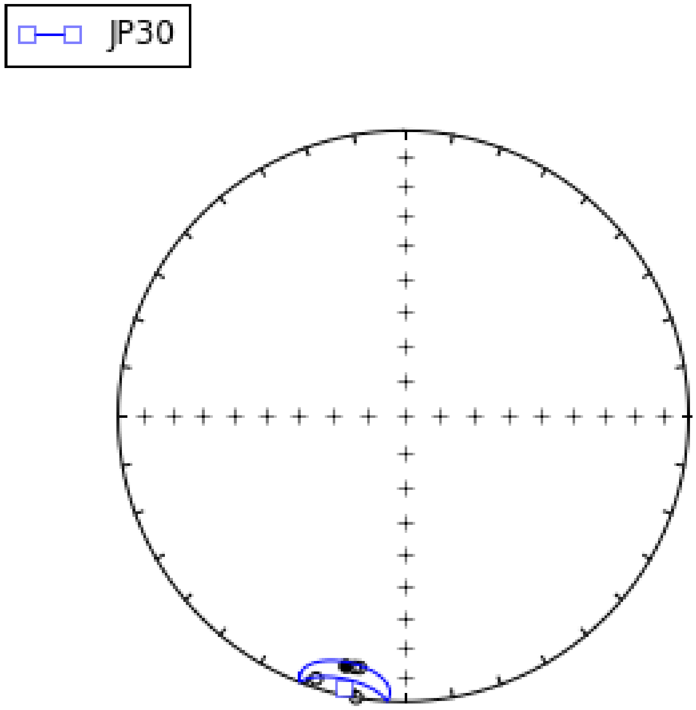
```
In [83]: JP30_dates = [1112.8, 1111.3, 1112.4, 1111.2,
                       1111.5, 1112.2, 1111.1, 1112.1]
         JP30_1sigma = [0.7, 0.75, 0.5, 0.85, 1.2, 0.65, 0.85, 1.5]
         weighted_mean(JP30_dates,JP30_1sigma)
```

```
The weighted mean is:
1111.98622406
```

```
With a 2sigma error of:
0.532765107538
```

In [84]:
```
JP30= Pancake_DF_tc.ix[Pancake_DF_tc['er_site_name'] == 'JP30']
JP30.reset_index(inplace=True)

JP30_tc_directions=[]
for n in range(len(JP30)):
    Dec,Inc=JP30['specimen_dec'][n],JP30['specimen_inc'][n]
    JP30_tc_directions.append([Dec,Inc,1.])

JP30_tc_mean=pmag.fisher_mean(JP30_tc_directions)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(fignum)
IPmag.iplotDI(JP30_tc_directions)
IPmag.iplotDImean(JP30_tc_mean['dec'],JP30_tc_mean['inc'],
                  JP30_tc_mean["alpha95"],color='b',marker='s',
                  label='JP30')
JP30_tc_mean
```

Out[84]:
```
{'alpha95': 9.5144858714743155,
 'csd': 9.9985743330743801,
 'dec': 192.30939984396562,
 'inc': -3.142080997210154,
 'k': 65.628711602779887,
 'n': 5,
 'r': 4.9390510661825857}
```



The JP30 data are very consistent. We add this site to the data compilation.

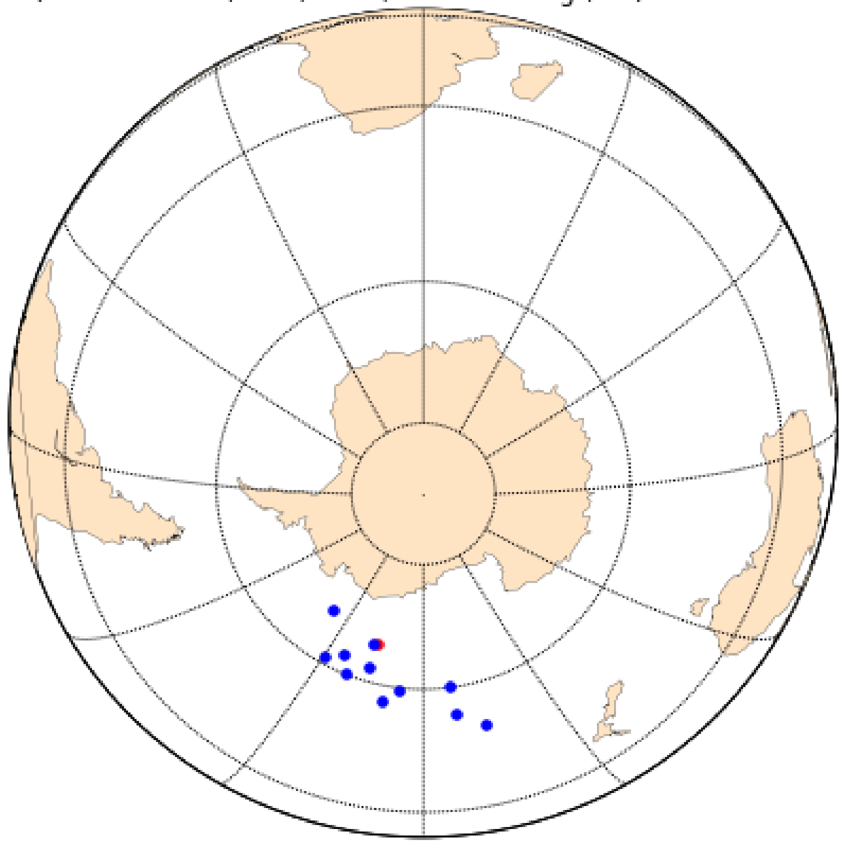
```
In [85]: #Add new data points to cooling unit table
         cooling_unit_means.loc['JP30'] = pd.Series({'site_ID':'JP30',
                                                     'site_lat':-22.70,
                                                     'site_long':26.61,
                                                     'n':int(JP30_tc_mean['n']),
                                                     'dec_tc':round(JP30_tc_mean['dec'],1),
                                                     'inc_tc':round(JP30_tc_mean['inc'],2),
                                                     'a_95':round(JP30_tc_mean['alpha95'],1),
                                                     'k':round(JP30_tc_mean['k'],1),
                                                     'date':1112.0,
                                                     'date_error':0.5})
```

2.5.9 Seidel (2004) and Pancake (2001) VGP calculation

Virtual geomagnetic poles can be calculated from these mean directions using the site location and mean declination and inclination. These VGPs are calculated below. Some of these VGPs will change as the data are integrated with our new results.

```
In [86]: IPmag.VGP_calc(cooling_unit_means)
```

```
In [87]: #create basemap for VGP plot
         plt.figure(figsize=(7, 7))
         m1 = Basemap(projection=’ortho’,lat_0=-80,lon_0=30,resolution=’c’,
                      area_thresh=50000)
         m1.drawcoastlines(linewidth=0.25)
         m1.fillcontinents(color=’bisque’,lake_color=’white’,zorder=1)
         m1.drawmapboundary(fill_color=’white’)
         m1.drawmeridians(np.arange(0,360,30))
         m1.drawparallels(np.arange(-90,90,30))

         for n in range(len(cooling_unit_means)):
             if cooling_unit_means[’pole_lat’][n] < 0:
                 IPmag.poleplot(m1,cooling_unit_means[’pole_long’][n],
                               cooling_unit_means[’pole_lat’][n],0,color=’b’)
             else:
                 IPmag.poleplot(m1,cooling_unit_means[’pole_long_rev’][n],
                               cooling_unit_means[’pole_lat_rev’][n],0,color=’r’)
         plt.title(’Seidel (2004) and Pancake (2001) VGPs, north-seeking (red) and south-seeki
         plt.legend(loc=1)
         plt.show()
```



2.6 Data from Jones and McElhinny (1966)

In the introduction to this notebook, the rationale for what sites from Jones and McElhinny (1966) will be integrated into this compilation is explained. These sites are: 3, 7, 8, 10, 12, 13.

The data for these sites are from Table 2 of Jones and McElhinny (1966). Tilt-corrections for these sites were variably applied using the orientation of the overlying sediments as explained in this quotation:

> In the vicinity of site 10, the Waterberg red beds that have been intruded by the dolerite sill, dip at 24 toward the north (5 east of north). In this region the similar folding of the overlying Karroo sediments proves that the age of the folding was post-Karroo. The data presented in Table 2 and Figures 2 and 3 relating to this site have therefore been corrected for the attitude of the adjacent sediments. At the other sites, the intruded beds were either fiat lying or dipped at not more than 10 in the east-west direction. Since the directions of magnetization are nearly horizontal in a

north-south direction, the correction for the attitude of the adjacent sediments makes no significant difference to the directions of magnetizations. Sites 1 to 13 include all the post-Waterberg diabases from which results were obtained and some diabases which have been classified as post-Transvaal or post-Ventersdorp.

```
In [88]: J_M66 = Gose06_data_table.ix[Gose06_data_table['Ref'] ==
                                      'Jones and McElhinny [1966]']
         J_M66.set_index(['Site ID'],drop='True',inplace='True')
         J_M66
```

Out[88]:

	Site Lat (N)	Site Long (E)	N	R	Dec	Inc	k	a95	Lat	Long	dp	dm	Ref
Site ID													
J_M7	-24.33	26.13	6	5.970	193.5	-5.5	165.0	5.2	59.9	54.2	NaN	NaN	Jones and McElhinny [1966]
J_M8	-24.23	25.87	7	6.545	191.0	-33.0	13.2	17.0	46.5	41.3	NaN	NaN	Jones and McElhinny [1966]
J_M9	-24.67	25.88	6	5.987	187.5	-13.0	379.0	3.4	57.7	40.3	NaN	NaN	Jones and McElhinny [1966]
A1	-24.93	25.30	4	3.988	329.5	8.5	244.0	5.9	48.2	335.5	NaN	NaN	Jones and McElhinny [1966]
J_M1	-23.10	26.60	8	7.961	190.5	14.0	180.0	4.2	70.7	62.7	NaN	NaN	Jones and McElhinny [1966]
J_M2	-23.05	26.55	7	6.991	190.5	9.5	674.0	2.3	69.1	56.6	NaN	NaN	Jones and McElhinny [1966]
J_M3	-23.00	26.41	8	7.991	190.5	4.0	796.0	2.0	66.6	53.7	NaN	NaN	Jones and McElhinny [1966]
J_M4	-23.00	26.41	6	5.981	187.0	5.0	254.0	6.7	68.5	46.1	NaN	NaN	Jones and McElhinny [1966]
J_M5	-23.00	26.41	7	6.981	187.0	5.0	323.0	3.3	38.4	45.7	NaN	NaN	Jones and McElhinny [1966]
J_M6	-22.90	26.40	8	7.992	186.5	0.5	897.0	1.8	66.4	42.8	NaN	NaN	Jones and McElhinny [1966]
J_M10	-22.92	29.93	5	4.940	194.0	24.0	66.5	9.5	73.1	84.2	NaN	NaN	Jones and McElhinny [1966]
J_M11	-25.50	29.46	6	5.633	188.0	6.0	13.6	19.0	66.2	50.2	NaN	NaN	Jones and McElhinny [1966]
J_M12	-26.90	28.53	6	5.983	16.0	-14.5	292.0	3.9	65.3	69.3	NaN	NaN	Jones and McElhinny [1966]
J_M13	-25.70	28.53	10	9.878	183.0	-3.0	73.5	5.7	62.7	34.6	NaN	NaN	Jones and McElhinny [1966]

```
In [89]: #Add A1 to unknown intrusion table, because it is likely older
         unknown_intrusions.loc['J_M_A1'] = pd.Series({'site_ID':'J_M_A1',
                                     'site_lat':-24.93,
                                     'site_long':25.30,
                                     'n':int(J_M66['N']['A1']),
                                     'dec_tc':round(J_M66['Dec']['A1'],1),
                                     'inc_tc':round(J_M66['Inc']['A1'],1),
                                     'a_95':round(J_M66['a95']['A1'],1),
                                     'k':round(J_M66['k']['A1'],1)})
```

```
In [90]: J_M66_edit = J_M66
         J_M66_edit = J_M66_edit.drop('J_M1')
         J_M66_edit = J_M66_edit.drop('J_M2')
         J_M66_edit = J_M66_edit.drop('J_M4')
         J_M66_edit = J_M66_edit.drop('J_M5')
         J_M66_edit = J_M66_edit.drop('J_M6')
         J_M66_edit = J_M66_edit.drop('J_M9')
         J_M66_edit = J_M66_edit.drop('A1')
         J_M66_edit = J_M66_edit.drop('J_M11')
         J_M66_edit
```

Out[90]:

	Site Lat (N)	Site Long (E)	N	R	Dec	Inc	k	a95	Lat	Long	dp	dm	Ref
Site ID													
J_M7	-24.33	26.13	6	5.970	193.5	-5.5	165.0	5.2	59.9	54.2	NaN	NaN	Jones and McElhinny [1966]
J_M8	-24.23	25.87	7	6.545	191.0	-33.0	13.2	17.0	46.5	41.3	NaN	NaN	Jones and McElhinny [1966]
J_M3	-23.00	26.41	8	7.991	190.5	4.0	796.0	2.0	66.6	53.7	NaN	NaN	Jones and McElhinny [1966]
J_M10	-22.92	29.93	5	4.940	194.0	24.0	66.5	9.5	73.1	84.2	NaN	NaN	Jones and McElhinny [1966]
J_M12	-26.90	28.53	6	5.983	16.0	-14.5	292.0	3.9	65.3	69.3	NaN	NaN	Jones and McElhinny [1966]
J_M13	-25.70	28.53	10	9.878	183.0	-3.0	73.5	5.7	62.7	34.6	NaN	NaN	Jones and McElhinny [1966]

In [91]:
```
#calculate dp/dm for the Jones and McElhinny (1966) sites that are to be used
pi180=np.pi/180.
J_M66_edit['paleolatitude'] = 0.
for n in range(len(J_M66_edit)):
    Paleolat = np.arctan(0.50*np.tan(J_M66_edit['Inc'][n]*pi180))/pi180
    J_M66_edit['paleolatitude'][n] = np.round_(Paleolat, 1)
    dp = (1.+3.*(np.cos(pi180*(90.-Paleolat)))**2.)*J_M66_edit['a95'][n]/2.
    J_M66_edit['dp'][n] = np.round_(dp, 1)
    dm = J_M66_edit['a95'][n]*np.sin(pi180*(90.-Paleolat))/np.cos(
        pi180*J_M66_edit['Inc'][n])
    J_M66_edit['dm'][n] = np.round_(dm, 1)
J_M66_edit
```

```
/Users/Laurentia/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-packages/IPython/kernel
A value is trying to be set on a copy of a slice from a DataFrame

See the the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing
/Users/Laurentia/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-packages/IPython/kernel
A value is trying to be set on a copy of a slice from a DataFrame

See the the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing
/Users/Laurentia/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-packages/IPython/kernel
A value is trying to be set on a copy of a slice from a DataFrame

See the the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing
```

Out[91]:

	Site Lat (N)	Site Long (E)	N	R	Dec	Inc	k	a95	Lat	Long	dp	dm	Ref	paleolatit
Site ID														
J_M7	-24.33	26.13	6	5.970	193.5	-5.5	165.0	5.2	59.9	54.2	2.6	5.2	Jones and McElhinny [1966]	
J_M8	-24.23	25.87	7	6.545	191.0	-33.0	13.2	17.0	46.5	41.3	10.9	19.3	Jones and McElhinny [1966]	-
J_M3	-23.00	26.41	8	7.991	190.5	4.0	796.0	2.0	66.6	53.7	1.0	2.0	Jones and McElhinny [1966]	
J_M10	-22.92	29.93	5	4.940	194.0	24.0	66.5	9.5	73.1	84.2	5.4	10.2	Jones and McElhinny [1966]	
J_M12	-26.90	28.53	6	5.983	16.0	-14.5	292.0	3.9	65.3	69.3	2.0	4.0	Jones and McElhinny [1966]	
J_M13	-25.70	28.53	10	9.878	183.0	-3.0	73.5	5.7	62.7	34.6	2.9	5.7	Jones and McElhinny [1966]	

In [92]:
```
#Add new data points to cooling unit table
for n in range(len(J_M66_edit)):
    if J_M66_edit['Dec'][n] > 270 or J_M66_edit['Dec'][n] < 90:
        cooling_unit_means.loc[J_M66_edit.index[n]] = pd.Series({
                'cooling_unit_name':J_M66_edit.index[n],
```

```
                  'site_ID':J_M66_edit.index[n],
                  'site_lat':J_M66_edit['Site Lat (N)'][n],
                  'site_long':J_M66_edit['Site Long (E)'][n],
                  'n':J_M66_edit['N'][n],
                  'dec_tc':J_M66_edit['Dec'][n],
                  'inc_tc':J_M66_edit['Inc'][n],
                  'a_95':J_M66_edit['a95'][n],
                  'k':J_M66_edit['k'][n],
                  'paleolatitude':J_M66_edit['paleolatitude'][n],
                  'pole_lat':round(J_M66_edit['Lat'][n],1),
                  'pole_long':round(J_M66_edit['Long'][n],1),
                  'pole_lat_rev':-round(J_M66_edit['Lat'][n],1),
                  'pole_long_rev':round(J_M66_edit['Long'][n],1)+180})
    if J_M66_edit['Dec'][n] < 270 and J_M66_edit['Dec'][n] > 90:
        cooling_unit_means.loc[J_M66_edit.index[n]] = pd.Series({
                  'cooling_unit_name':J_M66_edit.index[n],
                  'site_ID':J_M66_edit.index[n],
                  'site_lat':J_M66_edit['Site Lat (N)'][n],
                  'site_long':J_M66_edit['Site Long (E)'][n],
                  'n':J_M66_edit['N'][n],
                  'dec_tc':J_M66_edit['Dec'][n],
                  'inc_tc':J_M66_edit['Inc'][n],
                  'a_95':J_M66_edit['a95'][n],
                  'k':J_M66_edit['k'][n],
                  'paleolatitude':J_M66_edit['paleolatitude'][n],
                  'pole_lat_rev':round(J_M66_edit['Lat'][n],1),
                  'pole_long_rev':round(J_M66_edit['Long'][n],1),
                  'pole_lat':-round(J_M66_edit['Lat'][n],1),
                  'pole_long':round(J_M66_edit['Long'][n],1)+180})
cooling_unit_means.ix['J_M12','date']=1108.5
cooling_unit_means.ix['J_M12','date_error']=0.8
```

2.7 Data from McElhinny and Opdyke (1964)

Next we add data from other individual cooling units calculated by McElhinny and Opdyke (1964). All of the sites studied are distinct cooling units except for sites B and C, which are from the same cooling unit. We exclude site C because it has lower precision than B.

Tilt-corrections for these data are small and already applied to data given in Table 1 of McElhinny and Opdyke (1964). The description from that paper is as follows:

> The Umkondo dolerites are not as widespread as the Mashonaland dolerites, but they occur over nearly 3 of latitude, and it seems likely that the ten sampling sites cover a sufficient time to average out the effects of secular variation. The dolerites intrude the Umkondo beds, which usually dip gently at about 5 in a southeasterly direction. The

directions of magnetization have therefore been corrected in each case for the attitude of the adjacent sediments at each sampling site. The corrections are very small; Fisher's k for the directions changes from 57 to 58 on application of the corrections. This change is not significant, but the corrections are so small that this is to be expected. The directions cited for the various Umkondo dolerite sites in Table 1 refer to the directions after correction for the attitude of the sampling sites.

```
In [93]: M_064_ALL = Gose06_data_table.ix[Gose06_data_table['Ref'] ==
                                          'McElhinny and Opdyke [1964]']
         #slice dataframe including certain sites - excl. repeated intrusion, site C
         M_064_ALL.set_index(['Site ID'],drop='True',inplace='True')
         M_064_ALL
```

Out[93]:

	Site Lat (N)	Site Long (E)	N	R	Dec	Inc	k	a95	Lat	Long	dp	dm	Ref
Site ID													
M_O_A	-18.00	32.80	9	8.757	186.0	-17.0	33.0	9.0	63.0	49.5	NaN	NaN	McElhinny and Opdyke [1964]
M_O_B	-18.10	32.90	5	4.985	171.5	-10.0	267.0	4.5	65.5	12.0	NaN	NaN	McElhinny and Opdyke [1964]
M_O_C	-18.20	32.85	8	7.671	184.5	-8.0	21.0	12.5	67.5	44.5	NaN	NaN	McElhinny and Opdyke [1964]
M_O_D	-18.45	32.76	10	9.288	168.0	-5.5	12.6	14.0	66.0	21.0	NaN	NaN	McElhinny and Opdyke [1964]
M_O_E	-19.53	32.63	10	9.902	185.0	-3.5	92.0	5.0	68.0	46.0	NaN	NaN	McElhinny and Opdyke [1964]
M_O_F	-19.60	32.80	8	7.966	179.5	-13.0	206.0	4.0	64.0	31.5	NaN	NaN	McElhinny and Opdyke [1964]
M_O_H	-19.85	32.95	10	9.576	185.0	-2.5	21.0	10.5	68.5	46.5	NaN	NaN	McElhinny and Opdyke [1964]
M_O_I	-19.90	32.80	8	6.755	176.0	-14.0	5.7	25.5	62.5	24.0	NaN	NaN	McElhinny and Opdyke [1964]
M_O_J	-20.53	32.66	7	6.570	180.5	-10.0	14.0	16.5	64.5	34.0	NaN	NaN	McElhinny and Opdyke [1964]

```
In [94]: M_064_edit = M_064_ALL
         M_064_edit = M_064_edit.drop('M_O_A')
         M_064_edit = M_064_edit.drop('M_O_C')
         M_064_edit = M_064_edit.drop('M_O_I')
```

```
In [95]: #calculate dp/dm and paleolatitude for the McElhinny and...
         #...Opdyke (1964) sites that are to be used
         pi180=np.pi/180.
         M_064_edit['paleolatitude'] = 0.
         for n in range(len(M_064_edit)):
             Paleolat = np.arctan(0.50*np.tan(M_064_edit['Inc'][n]*pi180))/pi180
             M_064_edit['paleolatitude'][n] = np.round_(Paleolat, 1)
             dp = (1.+3.*(np.cos(pi180*(90.-Paleolat)))**2.)*M_064_edit['a95'][n]/2.
             M_064_edit['dp'][n] = np.round_(dp, 1)
             dm = M_064_edit['a95'][n]*np.sin(pi180*(90.-Paleolat))/np.cos(
                 pi180*M_064_edit['Inc'][n])
             M_064_edit['dm'][n] = np.round_(dm, 1)
         M_064_edit
```

```
/Users/Laurentia/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-packages/IPython/kernel
A value is trying to be set on a copy of a slice from a DataFrame
```

```
See the the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing
/Users/Laurentia/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-packages/IPython/kernel
A value is trying to be set on a copy of a slice from a DataFrame

See the the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing
/Users/Laurentia/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-packages/IPython/kernel
A value is trying to be set on a copy of a slice from a DataFrame

See the the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing
```

Out[95]:

	Site Lat (N)	Site Long (E)	N	R	Dec	Inc	k	a95	Lat	Long	dp	dm	Ref	paleolat
Site ID														
M_O_B	-18.10	32.90	5	4.985	171.5	-10.0	267.0	4.5	65.5	12.0	2.3	4.6	McElhinny and Opdyke [1964]	
M_O_D	-18.45	32.76	10	9.288	168.0	-5.5	12.6	14.0	66.0	21.0	7.0	14.0	McElhinny and Opdyke [1964]	
M_O_E	-19.53	32.63	10	9.902	185.0	-3.5	92.0	5.0	68.0	46.0	2.5	5.0	McElhinny and Opdyke [1964]	
M_O_F	-19.60	32.80	8	7.966	179.5	-13.0	206.0	4.0	64.0	31.5	2.1	4.1	McElhinny and Opdyke [1964]	
M_O_H	-19.85	32.95	10	9.576	185.0	-2.5	21.0	10.5	68.5	46.5	5.3	10.5	McElhinny and Opdyke [1964]	
M_O_J	-20.53	32.66	7	6.570	180.5	-10.0	14.0	16.5	64.5	34.0	8.4	16.7	McElhinny and Opdyke [1964]	

```
In [96]: for n in range(len(M_064_edit)):
             cooling_unit_means.loc[M_064_edit.index[n]] = pd.Series({
                      'cooling_unit_name':M_064_edit.index[n],
                      'site_ID':M_064_edit.index[n],
                      'site_lat':M_064_edit['Site Lat (N)'][n],
                      'site_long':M_064_edit['Site Long (E)'][n],
                      'n':M_064_edit['N'][n],
                      'dec_tc':M_064_edit['Dec'][n],
                      'inc_tc':M_064_edit['Inc'][n],
                      'a_95':M_064_edit['a95'][n],
                      'k':M_064_edit['k'][n],
                      'paleolatitude':M_064_edit['paleolatitude'][n],
                      'pole_lat_rev':round(M_064_edit['Lat'][n],1),
                      'pole_long_rev':round(M_064_edit['Long'][n],1),
                      'pole_lat':-round(M_064_edit['Lat'][n],1),
                      'pole_long':round(M_064_edit['Long'][n],1)+180})
```

2.8 Data from Gose et al. (2006)

Next we add the remaining data first presented in Gose et al. (2006), which includes a set of sites from northeastern South Africa.

- WD1 is from a unique cooling unit
- WD8 is from a unique cooling unit
- WD17 and WD18 are from the same sill. WD17 has better statistics and data from that sampling locality are used here.

- WD19 is from a unique cooling unit
- WD25 is from a unique cooling unit
- WD26 is from a unique cooling unit
- WD32 and WD33 may or may not be from the same sill given the complex geometry of the dolerites and overall poor exposure in the area. The sites have distinct directions and we are consider them to be distinct sills for the purpose of this compilation.
- WD34 is from a unique cooling unit

We exclude NB2 given the interpretation of it as a Paleoproterozoic cooling unit and TG05B given that it is associated with a baked-contact test.

```
In [97]: G06_ALL=Gose06_data_table.ix[Gose06_data_table['Ref'] == 'this study Gose+06']
         #slice dataframe to include only certain sites -- see above description
         G06_ALL.set_index(['Site ID'],drop='True',inplace='True')
         G06_ALL
```

Out[97]:

	Site Lat (N)	Site Long (E)	N	R	Dec	Inc	k	a95	Lat	Long	dp	dm	Ref
Site ID													
WD1	-23.81	28.74	9	8.490	184.0	-8.5	15.7	13.4	61.7	37.3	6.8	13.5	this study Gose+06
WD8	-24.28	28.71	12	11.489	171.4	-26.3	21.5	9.6	50.9	15.4	5.6	10.4	this study Gose+06
WD17	-23.15	28.75	10	9.664	189.5	-18.8	26.8	9.5	55.9	45.6	5.2	9.9	this study Gose+06
WD18	-23.15	28.75	5	4.817	190.3	-11.5	21.8	16.8	59.3	49.1	8.6	17.0	this study Gose+06
WD19	-23.16	26.68	10	9.402	190.5	-43.5	15.1	12.9	40.4	41.2	10.0	16.0	this study Gose+06
WD25	-23.42	28.65	8	7.013	205.6	11.9	7.1	22.4	59.9	87.4	11.5	22.7	this study Gose+06
WD26	-23.95	28.39	13	12.456	171.7	10.6	22.1	9.0	69.8	4.0	4.6	9.1	this study Gose+06
WD32	-24.14	27.41	6	5.657	181.4	3.7	14.6	18.1	67.7	31.2	9.1	18.2	this study Gose+06
WD33	-24.05	27.32	10	9.218	206.9	-36.2	11.5	14.9	38.7	60.3	10.1	17.3	this study Gose+06
WD34	-23.84	26.93	7	6.560	158.6	-27.2	13.6	16.9	46.3	355.9	10.0	18.5	this study Gose+06
TG05B	-24.55	31.35	12/2	11.479	326.9	4.4	21.1	4.4	50.8	39.7	5.6	11.2	this study Gose+06
NB2	-22.56	30.86	8/3	7.602	173.4	-53.7	17.6	13.6	32.9	24.4	13.3	19.0	this study Gose+06

```
In [98]: #Add WD28H and NB2 to unknown site table, because it is likely older
         hansonmisc = Gose06_data_table.ix[Gose06_data_table['Ref'] ==
                                          'Hanson et al. [2004b]']
         #slice dataframe to include only certain sites -- see above description
         hansonmisc.set_index(['Site ID'],drop='True',inplace='True')
         unknown_intrusions.loc['WD28H'] = pd.Series({'site_ID':'WD28H',
                                    'site_lat':-24.5,
                                    'site_long':27.56,
                                    'n':hansonmisc['N']['WD28H'],
                                    'dec_tc':round(hansonmisc['Dec']['WD28H'],1),
                                    'inc_tc':round(hansonmisc['Inc']['WD28H'],1),
                                    'a_95':round(hansonmisc['a95']['WD28H'],1),
                                    'k':round(hansonmisc['k']['WD28H'],1)})

         hansonmisc2 = Gose06_data_table.ix[Gose06_data_table['Site ID'] ==
```

```
                                           'L-2']
#slice dataframe to include only certain sites -- see above description
hansonmisc2.set_index(['Site ID'],drop='True',inplace='True')
unknown_intrusions.loc['L-2'] = pd.Series({'site_ID':'L-2',
                               'site_lat':-25.6,
                               'site_long':29.62,
                               'n':11,
                               'dec_tc':round(hansonmisc2['Dec']['L-2'],1),
                               'inc_tc':round(hansonmisc2['Inc']['L-2'],1),
                               'a_95':round(hansonmisc2['a95']['L-2'],1),
                               'k':round(hansonmisc2['k']['L-2'],1)})

unknown_intrusions.loc['NB2'] = pd.Series({'site_ID':'NB2',
                               'site_lat':-22.56,
                               'site_long':30.86,
                               'n':8,
                               'dec_tc':round(G06_ALL['Dec']['NB2'],1),
                               'inc_tc':round(G06_ALL['Inc']['NB2'],1),
                               'a_95':round(G06_ALL['a95']['NB2'],1),
                               'k':round(G06_ALL['k']['NB2'],1)})
```

```
In [99]: G06_edit = G06_ALL
         G06_edit = G06_edit.drop('WD18')
         G06_edit = G06_edit.drop('NB2')
         G06_edit = G06_edit.drop('TG05B')
         G06_edit
```

Out[99]:

	Site Lat (N)	Site Long (E)	N	R	Dec	Inc	k	a95	Lat	Long	dp	dm	Ref
Site ID													
WD1	-23.81	28.74	9	8.490	184.0	-8.5	15.7	13.4	61.7	37.3	6.8	13.5	this study Gose+06
WD8	-24.28	28.71	12	11.489	171.4	-26.3	21.5	9.6	50.9	15.4	5.6	10.4	this study Gose+06
WD17	-23.15	28.75	10	9.664	189.5	-18.8	26.8	9.5	55.9	45.6	5.2	9.9	this study Gose+06
WD19	-23.16	26.68	10	9.402	190.5	-43.5	15.1	12.9	40.4	41.2	10.0	16.0	this study Gose+06
WD25	-23.42	28.65	8	7.013	205.6	11.9	7.1	22.4	59.9	87.4	11.5	22.7	this study Gose+06
WD26	-23.95	28.39	13	12.456	171.7	10.6	22.1	9.0	69.8	4.0	4.6	9.1	this study Gose+06
WD32	-24.14	27.41	6	5.657	181.4	3.7	14.6	18.1	67.7	31.2	9.1	18.2	this study Gose+06
WD33	-24.05	27.32	10	9.218	206.9	-36.2	11.5	14.9	38.7	60.3	10.1	17.3	this study Gose+06
WD34	-23.84	26.93	7	6.560	158.6	-27.2	13.6	16.9	46.3	355.9	10.0	18.5	this study Gose+06

```
In [100]: #calculate paleolatitude for the NEW Gose et al. (2006) sites that are to be used
          pi180=np.pi/180.
          G06_edit['paleolatitude'] = 0.
          for n in range(len(G06_edit)):
              Paleolat = np.arctan(0.50*np.tan(G06_edit['Inc'][n]*pi180))/pi180
              G06_edit['paleolatitude'][n] = np.round_(Paleolat, 1)
          G06_edit
```

```
/Users/Laurentia/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-packages/IPython/kernel
```

```
A value is trying to be set on a copy of a slice from a DataFrame

See the the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing
```

Out[100]:

	Site Lat (N)	Site Long (E)	N	R	Dec	Inc	k	a95	Lat	Long	dp	dm	Ref	paleolatitude
Site ID														
WD1	-23.81	28.74	9	8.490	184.0	-8.5	15.7	13.4	61.7	37.3	6.8	13.5	this study Gose+06	-4.3
WD8	-24.28	28.71	12	11.489	171.4	-26.3	21.5	9.6	50.9	15.4	5.6	10.4	this study Gose+06	-13.9
WD17	-23.15	28.75	10	9.664	189.5	-18.8	26.8	9.5	55.9	45.6	5.2	9.9	this study Gose+06	-9.7
WD19	-23.16	26.68	10	9.402	190.5	-43.5	15.1	12.9	40.4	41.2	10.0	16.0	this study Gose+06	-25.4
WD25	-23.42	28.65	8	7.013	205.6	11.9	7.1	22.4	59.9	87.4	11.5	22.7	this study Gose+06	6.0
WD26	-23.95	28.39	13	12.456	171.7	10.6	22.1	9.0	69.8	4.0	4.6	9.1	this study Gose+06	5.3
WD32	-24.14	27.41	6	5.657	181.4	3.7	14.6	18.1	67.7	31.2	9.1	18.2	this study Gose+06	1.9
WD33	-24.05	27.32	10	9.218	206.9	-36.2	11.5	14.9	38.7	60.3	10.1	17.3	this study Gose+06	-20.1
WD34	-23.84	26.93	7	6.560	158.6	-27.2	13.6	16.9	46.3	355.9	10.0	18.5	this study Gose+06	-14.4

```
In [101]: for n in range(len(G06_edit)):
              cooling_unit_means.loc[G06_edit.index[n]] = pd.Series({
                       'cooling_unit_name':G06_edit.index[n],
                       'site_ID':G06_edit.index[n],
                       'site_lat':G06_edit['Site Lat (N)'][n],
                       'site_long':G06_edit['Site Long (E)'][n],
                       'n':G06_edit['N'][n],
                       'dec_tc':G06_edit['Dec'][n],
                       'inc_tc':G06_edit['Inc'][n],
                       'a_95':G06_edit['a95'][n],
                       'k':G06_edit['k'][n],
                       'paleolatitude':G06_edit['paleolatitude'][n],
                       'pole_lat_rev':round(G06_edit['Lat'][n],1),
                       'pole_long_rev':round(G06_edit['Long'][n],1),
                       'pole_lat':-round(G06_edit['Lat'][n],1),
                       'pole_long':round(G06_edit['Long'][n],1)+180})
          cooling_unit_means['pole_long']['WD34'] = round(G06_edit['Long']['WD34'],1)-180
```

```
/Users/Laurentia/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-packages/IPython/kernel
A value is trying to be set on a copy of a slice from a DataFrame

See the the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing
```

```
In [102]: cooling_unit_means
```

Out[102]:

	site_ID	site_lat	site_long	n	dec_tc	inc_tc	a_95	k	date	date_error	paleolatitude	pole_lat
W01_W02	W01_W02	-25.48	29.45	25	175.6	-18.40	6.2	22.6	NaN	NaN	-9.443434	-54.814751
W04	W04	-25.75	29.45	12	171.5	-22.30	4.9	80.8	NaN	NaN	-11.588697	-51.755125
W05	W05	-25.76	29.48	11	176.4	-7.80	7.5	38.0	NaN	NaN	-3.918154	-60.117314
W08_W09	W08_W09	-25.62	29.10	20	192.8	15.90	1.9	297.2	NaN	NaN	8.106047	-68.663589
VF1_VF2	VF1_VF2	-25.80	27.50	21	7.2	-6.80	3.1	103.2	1108.6	1.2	-3.412015	66.568944
TG-S-series	TG-S-series	-24.20	31.40	120	186.3	2.87	5.7	7.2	1111.5	0.4	1.435901	-66.433862
TG-N-series	TG-N-series	-23.20	31.20	13	182.8	-14.73	6.5	41.9	NaN	NaN	-7.488730	-59.189351
JP15	JP15	-24.32	25.53	8	195.5	-0.79	7.3	58.6	NaN	NaN	-0.395019	-61.075442
JP19	JP19	-24.23	25.64	5	188.2	-15.45	14.9	27.2	NaN	NaN	-7.868014	-56.919550
Mosolotsane 1	JP22, JP23, JP24	-22.91	26.39	15	188.9	-6.70	3.3	138.8	1109.3	0.6	-3.361491	-62.329826
Shoshong	JP26, JP31, JP33, JP34)	-23.00	26.48	25	192.5	-8.43	3.3	78.4	1109.3	0.4	-4.237935	-60.152547
JP30	JP30	-22.70	26.61	5	192.3	-3.14	9.5	65.6	1112.0	0.5	-1.571180	-62.928814
J_M7	J_M7	-24.33	26.13	6	193.5	-5.50	5.2	165.0	NaN	NaN	-2.800000	-59.900000
J_M8	J_M8	-24.23	25.87	7	191.0	-33.00	17.0	13.2	NaN	NaN	-18.000000	-46.500000
J_M3	J_M3	-23.00	26.41	8	190.5	4.00	2.0	796.0	NaN	NaN	2.000000	-66.600000
J_M10	J_M10	-22.92	29.93	5	194.0	24.00	9.5	66.5	NaN	NaN	12.600000	-73.100000
J_M12	J_M12	-26.90	28.53	6	16.0	-14.50	3.9	292.0	1108.5	0.8	-7.400000	65.300000
J_M13	J_M13	-25.70	28.53	10	183.0	-3.00	5.7	73.5	NaN	NaN	-1.500000	-62.700000
M_O_B	M_O_B	-18.10	32.90	5	171.5	-10.00	4.5	267.0	NaN	NaN	-5.000000	-65.500000
M_O_D	M_O_D	-18.45	32.76	10	168.0	-5.50	14.0	12.6	NaN	NaN	-2.800000	-66.000000
M_O_E	M_O_E	-19.53	32.63	10	185.0	-3.50	5.0	92.0	NaN	NaN	-1.800000	-68.000000
M_O_F	M_O_F	-19.60	32.80	8	179.5	-13.00	4.0	206.0	NaN	NaN	-6.600000	-64.000000
M_O_H	M_O_H	-19.85	32.95	10	185.0	-2.50	10.5	21.0	NaN	NaN	-1.300000	-68.500000
M_O_J	M_O_J	-20.53	32.66	7	180.5	-10.00	16.5	14.0	NaN	NaN	-5.000000	-64.500000
WD1	WD1	-23.81	28.74	9	184.0	-8.50	13.4	15.7	NaN	NaN	-4.300000	-61.700000
WD8	WD8	-24.28	28.71	12	171.4	-26.30	9.6	21.5	NaN	NaN	-13.900000	-50.900000
WD17	WD17	-23.15	28.75	10	189.5	-18.80	9.5	26.8	NaN	NaN	-9.700000	-55.900000
WD19	WD19	-23.16	26.68	10	190.5	-43.50	12.9	15.1	NaN	NaN	-25.400000	-40.400000
WD25	WD25	-23.42	28.65	8	205.6	11.90	22.4	7.1	NaN	NaN	6.000000	-59.900000
WD26	WD26	-23.95	28.39	13	171.7	10.60	9.0	22.1	NaN	NaN	5.300000	-69.800000
WD32	WD32	-24.14	27.41	6	181.4	3.70	18.1	14.6	NaN	NaN	1.900000	-67.700000
WD33	WD33	-24.05	27.32	10	206.9	-36.20	14.9	11.5	NaN	NaN	-20.100000	-38.700000
WD34	WD34	-23.84	26.93	7	158.6	-27.20	16.9	13.6	NaN	NaN	-14.400000	-46.300000

2.9 Data from Mare et al. (2006)

Next we add data not included in Gose et al. (2006), reported in Mare et al. (2006) from the Wilge River Formation dolerite. The sites sampled overlap with intrusions reported in Seidel (2004) and Gose et al. (2006), therefore only two new cooling unit means are used, WRD4 and WRD5, two others being very scattered.

Another intrusion is sampled in Mare et al. (2006), the Swaershoek Formation dolerite. However, data from this intrusion are not consistent and sample level data are not supplied, making it difficult to incorporate data from different sites into a correctly averaged mean. Given the large differences between site mean directions, we exclude these data from our analysis.

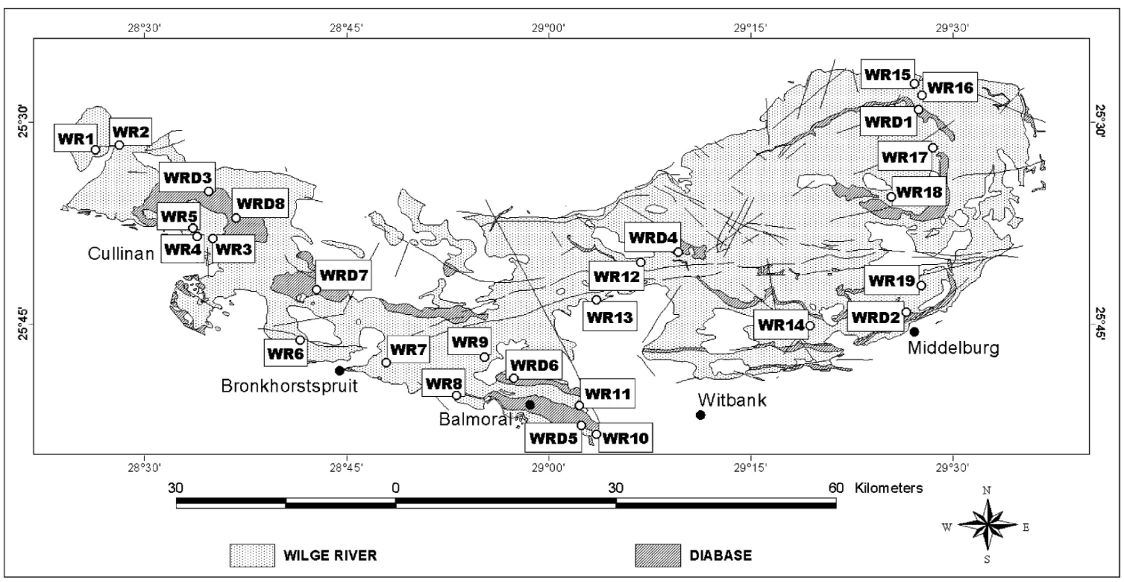
```
In [103]: Mare_WRD=pd.read_csv('../Data/Prior_Data/Mare+06.csv')
          pi180=np.pi/180.
          Mare_WRD['paleolatitude'] = 0.
          for i in range(len(Mare_WRD)):
              Paleolat = np.arctan(0.50*np.tan(Mare_WRD.loc[i,'I']*pi180))/pi180
              Mare_WRD.loc[i,'paleolatitude'] = np.abs(np.round_(Paleolat, 1))
          Mare_WRD
```

Out[103]:

	site_ID	Slat	Slon	N	D	I	a95	Plat	Plong	dp	dm	a_95	Ref	paleolatitude
0	WRD1	-25.49	29.46	6	177.3	-5.1	13.7	-61.8	203.7	6.9	13.7	9.7	Mare et al. [2006]	2.6
1	WRD2	-25.74	29.45	5	142.5	-10.2	16.6	-42.3	154.4	8.5	16.8	12.0	Mare et al. [2006]	5.1
2	WRD3	-25.59	28.58	4	174.0	9.5	6.3	-68.4	192.0	3.2	6.4	4.5	Mare et al. [2006]	4.8
3	WRD4	-25.66	29.16	5	178.1	11.1	8.2	-69.9	203.7	4.2	8.3	5.9	Mare et al. [2006]	5.6
4	WRD5	-25.88	29.03	5	173.9	15.2	15.3	-70.9	190.2	8.1	15.7	11.3	Mare et al. [2006]	7.7
5	WRD6	-25.82	28.95	8	201.7	33.6	33.6	-57.2	252.0	16.8	33.6	23.8	Mare et al. [2006]	0.6
6	WRD7	-25.71	28.71	8	185.1	21.7	22.7	-74.8	228.1	12.6	24.0	17.4	Mare et al. [2006]	11.3
7	WRD8	-25.62	28.61	7	205.5	29.6	16.8	-64.3	281.3	10.3	18.6	13.8	Mare et al. [2006]	15.9

Below is a site locality map shown from Mare et al. (2006) which can be compared to the the Seidel (2004) Middleburg area map (W- series) shown earlier in this supplement.

```
In [104]: Mare_WRD_sites=Image(filename=
                                    'Local_PNGs/Mare_WRD_site_map.png')
          display(Mare_WRD_sites)
```



We exclude sites that are redundant because there is no way to combine the sample level data and because they are consistent with results from Seidel (2004): 1. WRD1 comes from the same intrusion as W_02 and W_02 (Seidel, 2004) 2. WRD2 comes from the same intrusion as W_04 (Seidel, 2004) 3. WRD3 and WRD8 come from the same intruson as W_08 and W_09 (Seidel, 2004)

Therefore we include sites WRD4, WRD5, WRD6, and WRD7, although the latter two are likely to be excluded further in the analysis due to thier high error interval.

```
In [105]: for i in range(3,7):
                    cooling_unit_means.loc[Mare_WRD['site_ID'][i]] = pd.Series({
                         'cooling_unit_name':Mare_WRD['site_ID'][i],
                         'site_ID':Mare_WRD['site_ID'][i],
```

```
                'site_lat':Mare_WRD['Slat'][i],
                'site_long':Mare_WRD['Slon'][i],
                'n':Mare_WRD['N'][i],
                'dec_tc':Mare_WRD['D'][i],
                'inc_tc':Mare_WRD['I'][i],
                'a_95':Mare_WRD['a95'][i],
                'paleolatitude':Mare_WRD['paleolatitude'][i],
                'pole_lat_rev':-round(Mare_WRD['Plat'][i],1),
                'pole_long_rev':round(Mare_WRD['Plong'][i],1)-180,
                'pole_lat':round(Mare_WRD['Plat'][i],1),
                'pole_long':round(Mare_WRD['Plong'][i],1)})
```

2.10 Data from Wilson et al. (1987)

Next we add to the dataset additional data not included in Gose et al. (2006), reported in Wilson et al. (1987) from northern Zimbabwe. The two sites provided have a limited number of samples and poorly defined sampling locality (with unknown "Grid" coordinates). The primary reason that these intrusions are considered Umkondo-aged is because of thier paleomagnetic directions. Also there are volcanics to the east that are Umkondo in age, therefore it is very likely that the Umkondo intrusive system extended across the Zimbabwe craton.

```
In [106]: Wilson_Wil=pd.read_csv('../Data/Prior_Data/Wilson+87.csv')
          pi180=np.pi/180.
          Wilson_Wil['paleolatitude'] = 0.
          for i in range(len(Wilson_Wil)):
              Paleolat = np.arctan(0.50*np.tan(Wilson_Wil.loc[i,'I']*pi180))/pi180
              Wilson_Wil.loc[i,'paleolatitude'] = np.abs(np.round_(Paleolat, 1))
          Wilson_Wil
```

Out[106]:

	site_ID	Slat	Slon	N	D	I	a95	Plat	Plong	dp	dm	a_95	Ref	paleolatitude
0	Wil_1	-17.9	31.5	5	181.4	-15.4	7.9	-64.2	214.7	4.2	8.1	5.9	Wilson et al. [1987]	7.8
1	Wil_2	-17.4	30.1	7	10.6	9.3	11.6	65.6	56.4	5.9	11.7	8.4	Wilson et al. [1987]	4.7

```
In [107]: for i in range(0,2):
                cooling_unit_means.loc[Wilson_Wil['site_ID'][i]] = pd.Series({
                     'cooling_unit_name':Wilson_Wil['site_ID'][i],
                     'site_ID':Wilson_Wil['site_ID'][i],
                     'site_lat':Wilson_Wil['Slat'][i],
                     'site_long':Wilson_Wil['Slon'][i],
                     'n':Wilson_Wil['N'][i],
                     'dec_tc':Wilson_Wil['D'][i],
                     'inc_tc':Wilson_Wil['I'][i],
                     'a_95':Wilson_Wil['a95'][i],
```

```
                    'paleolatitude':Wilson_Wil['paleolatitude'][i],
                    'pole_lat_rev':-round(Wilson_Wil['Plat'][i],1),
                    'pole_long_rev':round(Wilson_Wil['Plong'][i],1)-180,
                    'pole_lat':round(Wilson_Wil['Plat'][i],1),
                    'pole_long':round(Wilson_Wil['Plong'][i],1)})

        cooling_unit_means['pole_long_rev']['Wil_2'] = Wilson_Wil['Plong'][1]+180
```

```
/Users/Laurentia/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-packages/IPython/kernel
A value is trying to be set on a copy of a slice from a DataFrame

See the the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing
```

2.11 Summary of paleomagnetic directions

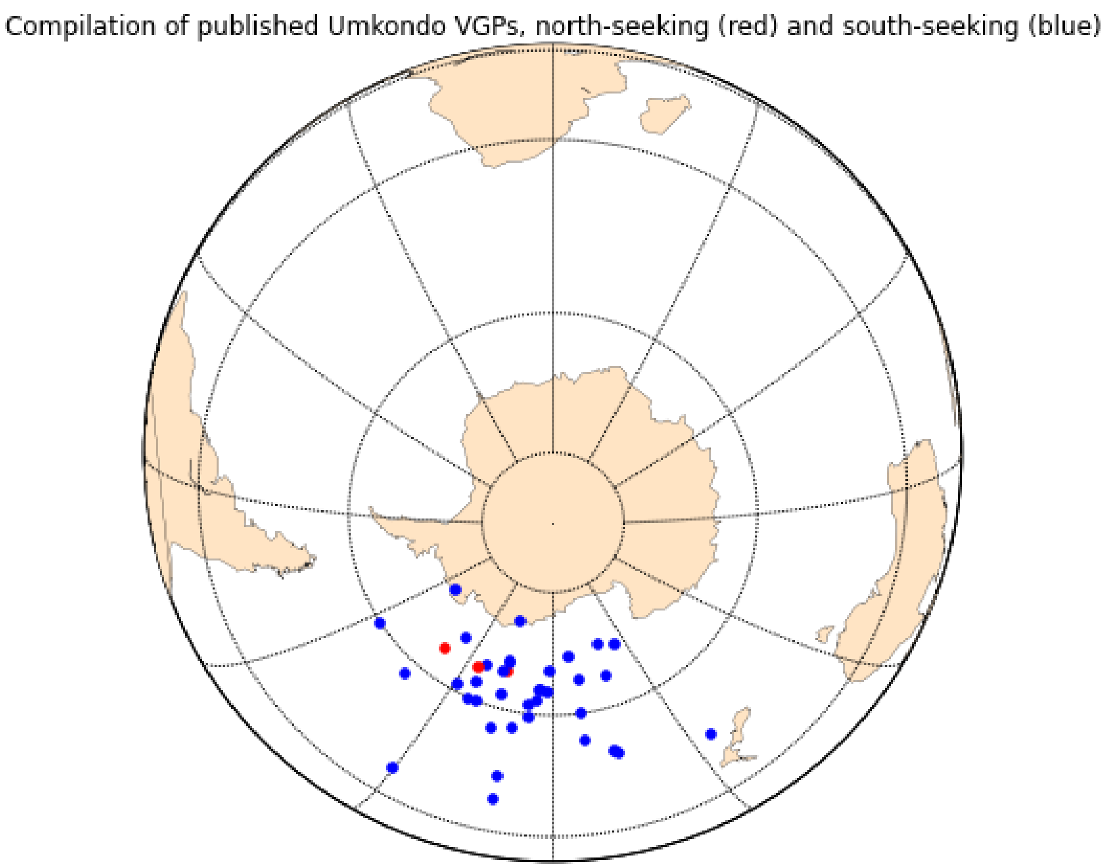
```
In [108]: cooling_unit_means
```

Out[108]:

	site_ID	site_lat	site_long	n	dec_tc	inc_tc	a_95	k	date	date_error	paleolatitude	pole_lat
W01_W02	W01_W02	-25.48	29.45	25	175.6	-18.40	6.2	22.6	NaN	NaN	-9.443434	-54.814751
W04	W04	-25.75	29.45	12	171.5	-22.30	4.9	80.8	NaN	NaN	-11.588697	-51.755125
W05	W05	-25.76	29.48	11	176.4	-7.80	7.5	38.0	NaN	NaN	-3.918154	-60.117314
W08_W09	W08_W09	-25.62	29.10	20	192.8	15.90	1.9	297.2	NaN	NaN	8.106047	-68.663589
VF1_VF2	VF1_VF2	-25.80	27.50	21	7.2	-6.80	3.1	103.2	1108.6	1.2	-3.412015	66.568944
TG-S-series	TG-S-series	-24.20	31.40	120	186.3	2.87	5.7	7.2	1111.5	0.4	1.435901	-66.433862
TG-N-series	TG-N-series	-23.20	31.20	13	182.8	-14.73	6.5	41.9	NaN	NaN	-7.488730	-59.189351
JP15	JP15	-24.32	25.53	8	195.5	-0.79	7.3	58.6	NaN	NaN	-0.395019	-61.075442
JP19	JP19	-24.23	25.64	5	188.2	-15.45	14.9	27.2	NaN	NaN	-7.868014	-56.919550
Mosolotsane 1	JP22, JP23, JP24	-22.91	26.39	15	188.9	-6.70	3.3	138.8	1109.3	0.6	-3.361491	-62.329826
Shoshong	JP26, JP31, JP33, JP34)	-23.00	26.48	25	192.5	-8.43	3.3	78.4	1109.3	0.4	-4.237935	-60.152547
JP30	JP30	-22.70	26.61	5	192.3	-3.14	9.5	65.6	1112.0	0.5	-1.571180	-62.928814
J_M7	J_M7	-24.33	26.13	6	193.5	-5.50	5.2	165.0	NaN	NaN	-2.800000	-59.900000
J_M8	J_M8	-24.23	25.87	7	191.0	-33.00	17.0	13.2	NaN	NaN	-18.000000	-46.500000
J_M3	J_M3	-23.00	26.41	8	190.5	4.00	2.0	796.0	NaN	NaN	2.000000	-66.600000
J_M10	J_M10	-22.92	29.93	5	194.0	24.00	9.5	66.5	NaN	NaN	12.600000	-73.100000
J_M12	J_M12	-26.90	28.53	6	16.0	-14.50	3.9	292.0	1108.5	0.8	-7.400000	65.300000
J_M13	J_M13	-25.70	28.53	10	183.0	-3.00	5.7	73.5	NaN	NaN	-1.500000	-62.700000
M_O_B	M_O_B	-18.10	32.90	5	171.5	-10.00	4.5	267.0	NaN	NaN	-5.000000	-65.500000
M_O_D	M_O_D	-18.45	32.76	10	168.0	-5.50	14.0	12.6	NaN	NaN	-2.800000	-66.000000
M_O_E	M_O_E	-19.53	32.63	10	185.0	-3.50	5.0	92.0	NaN	NaN	-1.800000	-68.000000
M_O_F	M_O_F	-19.60	32.80	8	179.5	-13.00	4.0	206.0	NaN	NaN	-6.600000	-64.000000
M_O_H	M_O_H	-19.85	32.95	10	185.0	-2.50	10.5	21.0	NaN	NaN	-1.300000	-68.500000
M_O_J	M_O_J	-20.53	32.66	7	180.5	-10.00	16.5	14.0	NaN	NaN	-5.000000	-64.500000
WD1	WD1	-23.81	28.74	9	184.0	-8.50	13.4	15.7	NaN	NaN	-4.300000	-61.700000
WD8	WD8	-24.28	28.71	12	171.4	-26.30	9.6	21.5	NaN	NaN	-13.900000	-50.900000
WD17	WD17	-23.15	28.75	10	189.5	-18.80	9.5	26.8	NaN	NaN	-9.700000	-55.900000
WD19	WD19	-23.16	26.68	10	190.5	-43.50	12.9	15.1	NaN	NaN	-25.400000	-40.400000
WD25	WD25	-23.42	28.65	8	205.6	11.90	22.4	7.1	NaN	NaN	6.000000	-59.900000
WD26	WD26	-23.95	28.39	13	171.7	10.60	9.0	22.1	NaN	NaN	5.300000	-69.800000
WD32	WD32	-24.14	27.41	6	181.4	3.70	18.1	14.6	NaN	NaN	1.900000	-67.700000
WD33	WD33	-24.05	27.32	10	206.9	-36.20	14.9	11.5	NaN	NaN	-20.100000	-38.700000
WD34	WD34	-23.84	26.93	7	158.6	-27.20	16.9	13.6	NaN	NaN	-14.400000	-46.300000
WRD4	WRD4	-25.66	29.16	5	178.1	11.10	8.2	NaN	NaN	NaN	5.600000	-69.900000
WRD5	WRD5	-25.88	29.03	5	173.9	15.20	15.3	NaN	NaN	NaN	7.700000	-70.900000
WRD6	WRD6	-25.82	28.95	8	201.7	1.10	33.6	NaN	NaN	NaN	0.600000	-57.200000
WRD7	WRD7	-25.71	28.71	8	185.1	21.70	22.7	NaN	NaN	NaN	11.300000	-74.800000
Wil_1	Wil_1	-17.90	31.50	5	181.4	-15.40	7.9	NaN	NaN	NaN	7.800000	-64.200000
Wil_2	Wil_2	-17.40	30.10	7	10.6	9.30	11.6	NaN	NaN	NaN	4.700000	65.600000

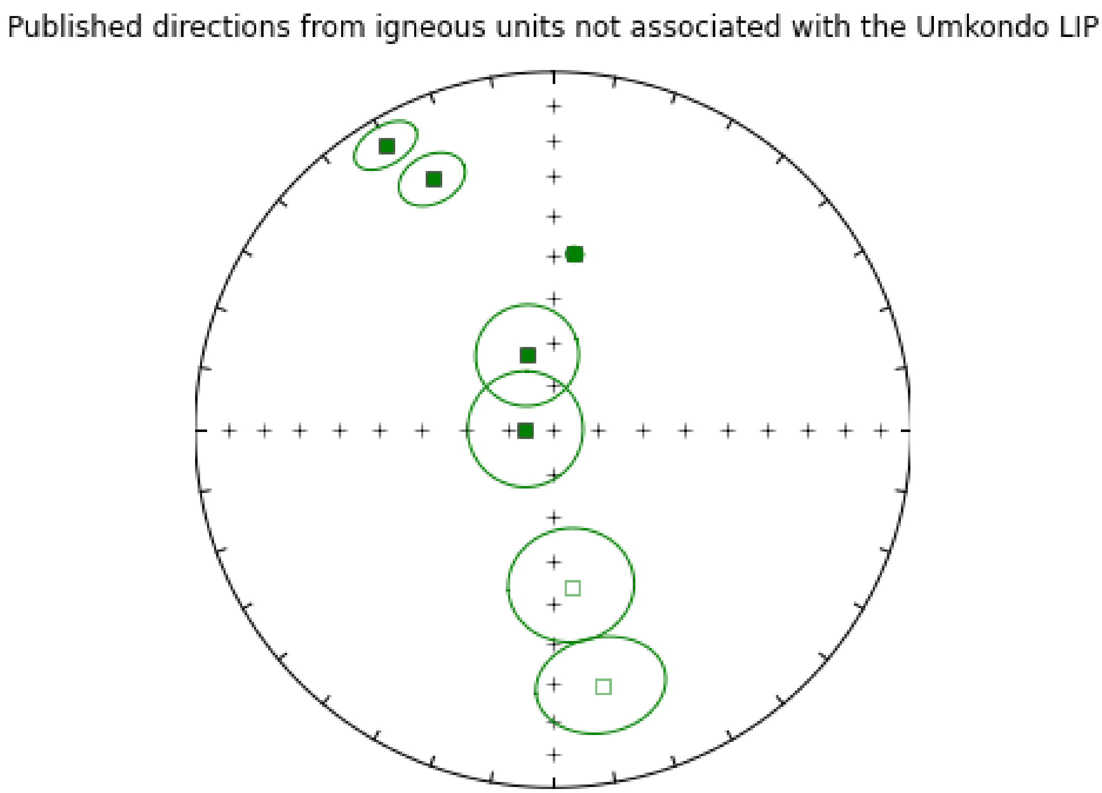
```
In [109]: #create basemap for VGP plot
          plt.figure(figsize=(7, 7))
          m1 = Basemap(projection='ortho',lat_0=-80,lon_0=30,resolution='c',
```

```
                area_thresh=50000)
m1.drawcoastlines(linewidth=0.25)
m1.fillcontinents(color='bisque',lake_color='white',zorder=1)
m1.drawmapboundary(fill_color='white')
m1.drawmeridians(np.arange(0,360,30))
m1.drawparallels(np.arange(-90,90,30))

for n in range(len(cooling_unit_means)):
    if cooling_unit_means['pole_lat'][n] < 0:
        IPmag.poleplot(m1,cooling_unit_means['pole_long'][n],
                       cooling_unit_means['pole_lat'][n],0,color='b')
    else:
        IPmag.poleplot(m1,cooling_unit_means['pole_long_rev'][n],
                       cooling_unit_means['pole_lat_rev'][n],0,color='r')
plt.title('Compilation of published Umkondo VGPs, north-seeking (red) and south-seek
plt.legend(loc=1)
plt.show()
```



Data from intrusions of unknown and Paleoproterozoic age are shown below, plotted on an equal-area plot. This compilation will be updated with additional data generated for this study.

In [110]:
```
IPmag.VGP_calc(unknown_intrusions)
fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
for n in range(len(unknown_intrusions)):
    IPmag.iplotDImean(unknown_intrusions['dec_tc'][n],
                      unknown_intrusions['inc_tc'][n],
                      unknown_intrusions['a_95'][n],
                      color='g',marker='s',label='')
plt.title('Published directions from igneous units not associated with the Umkondo LI
plt.show()
```



Pickle the data that needs to be used for calculation of integrated means with the newly developed data and the table of means

```
In [111]: #pickle sample data from the JP series that need to be...
          #...combined with new Botswana data
          pickle.dump(JP15_tc_dir_edited, open(
                  '../Data/Pickle/JP15','wb')) #Suping Sill
          pickle.dump(JP30_tc_directions, open(
                  '../Data/Pickle/JP30','wb')) #Mokgware Sill
          pickle.dump(JP26_31_33_34_tc_directions, open(
                  '../Data/Pickle/JP26_31_33_34','wb')) #Shosong sill
          pickle.dump(JP22_23_24_tc_directions, open(
                  '../Data/Pickle/JP22_23_24','wb')) #Mosolotsane 1 Sill
          #unknown or older intrusions
          pickle.dump(unknown_intrusions, open(
                  '../Data/Pickle/unknown_intrusions','wb'))
```

Drop the mean directions to which new data will be added through combination with the previous results and then pickle the cooling unit means table so that it can be used to calculate a new grand mean.

```
In [112]: cooling_unit_means_edit = cooling_unit_means

          cooling_unit_means_edit = cooling_unit_means_edit.drop('Mosolotsane 1')
          cooling_unit_means_edit = cooling_unit_means_edit.drop('Shoshong')
          cooling_unit_means_edit = cooling_unit_means_edit.drop('JP15')
          cooling_unit_means_edit = cooling_unit_means_edit.drop('JP30')

          pickle.dump(cooling_unit_means_edit, open(
                  '../Data/Pickle/cooling_unit_means_edit','wb'))
```

```
In [ ]:
```

3 Analysis of new paleomagnetic data from the Umkondo Large Igneous Province

A field season to Botswana in 2012 resulted in the collection of paleomagnetic samples from 32 mafic intrusions (some of them sampled in multiple locations) with the goal of developing new data from the Umkondo large igneous province (LIP). Paleomagnetic data for these intrusions were developed at the Institute for Rock Magnetism and in the new UC Berkeley paleomagnetism lab (as described within the main text).

3.1 Import needed modules for data analysis

```
In [1]: import pmag, pmagplotlib, IPmag
    import matplotlib.pyplot as plt
```

```
from mpl_toolkits.basemap import Basemap
import pandas as pd
pd.set_option('display.max_columns',500)
pd.set_option('display.max_rows',500)
from IPython.core.display import HTML
import numpy as np
import scipy as sp
from scipy import special
from IPython.display import display, Image
import pickle
%matplotlib inline

def _repr_latex_(self):
 return "\centering{%s}" % self.to_latex()
pd.DataFrame._repr_latex_ = _repr_latex_
```

3.2 Location of studied sites

Import and display table with location of studied sites.

```
In [2]: Site_Locations = pd.read_csv('../Data/Field_Data/BotswanaWaypoints.csv')
    Site_Locations
```

```
Out[2]:
```

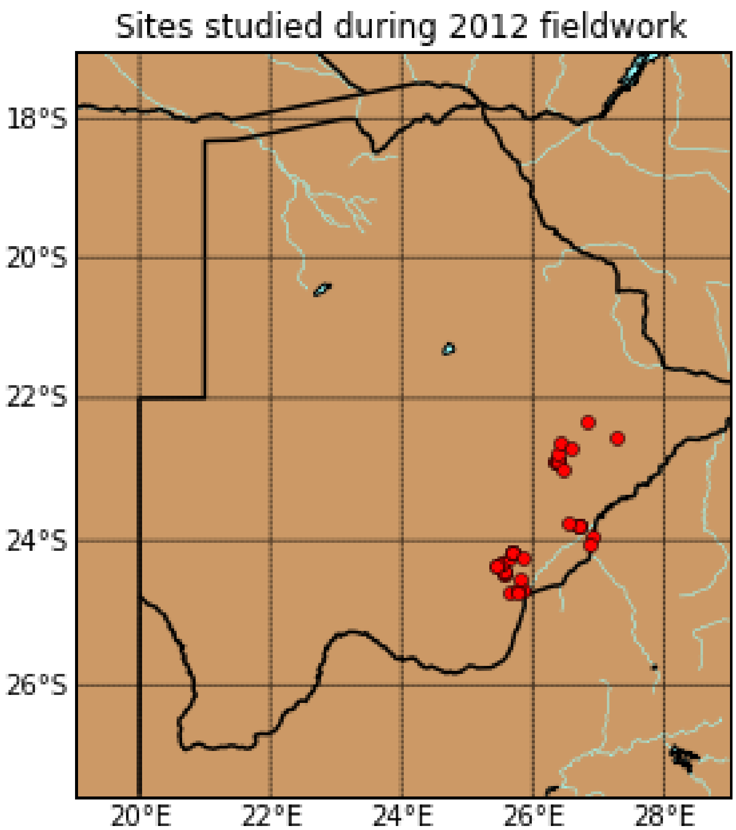
	ID	LAT(WGS84)	LONG(WGS84)	ELEV(meters)	COMMENTS
0	PW1	-24.68781	25.86215	0	same as JP1
1	PW2	-24.68822	25.86452	0	same as JP1
2	PW3	-24.74891	25.65574	0	NaN
3	PW4	-24.69056	25.77397	0	NaN
4	PW5	-24.72715	25.77590	0	NaN
5	PW6	-24.54694	25.80892	982	NaN
6	PW7	-24.47670	25.59706	1084	NaN
7	PW8	-24.47402	25.59705	1110	NaN
8	PW9	-24.45268	25.57415	1129	NaN
9	PW10	-24.41968	25.58463	1125	NaN
10	PW11	-24.32765	25.53224	1159	same as PW12
11	PW12	-24.32787	25.53375	1160	same as PW11
12	PW13	-24.25577	25.65073	1094	NaN
13	PW14	-24.18075	25.68963	1074	NaN
14	PW15	-24.18042	25.69191	1090	NaN
15	PW16	-24.18034	25.69156	1093	same sill as PW15 but not insitu
16	PW17	-24.18033	25.69155	1093	same sill as PW15 but not insitu
17	PW18	-24.37615	25.46877	0	PW18/19 are same sill
18	PW19	-24.37611	25.46841	0	PW18/19 are same sill
19	PW20	-24.24053	25.85217	1004	NaN
20	PW21	-22.90736	26.38935	1218	PW21/22 are same sill
21	PW22	-22.90699	26.38929	1215	PW21/22 are same sill
22	PW23	-22.90330	26.37027	1262	NaN
23	PW24	-22.89467	26.37410	1245	NaN
24	PW25	-22.89550	26.36726	1261	NaN
25	PW26	-22.89259	26.38113	1243	NaN
26	PW27	-22.89228	26.38196	1247	NaN
27	PW28	-23.00519	26.48383	1121	Shoshong sill
28	PW29	-22.77939	26.39372	1266	NaN
29	PW30	-22.64213	26.44260	1195	NaN
30	PW31	-22.70685	26.61142	1149	NaN
31	PW32	-22.33543	26.82285	1096	NaN
32	PW33	-22.57771	27.28736	1001	NaN
33	PW34	-23.81626	26.73769	976	NaN
34	PW35	-23.81403	26.73541	973	NaN
35	PW36	-23.81453	26.73503	973	NaN
36	PW37	-23.81408	26.73344	971	NaN
37	PW38	-23.78171	26.56308	975	NaN
38	PW39	-23.96123	26.91017	853	NaN
39	PW40	-24.04815	26.89136	854	NaN

3.2.1 Locality maps of sites

Location of sites within a map of all of Botswana and then a map zoomed in on the southeast part of the country.

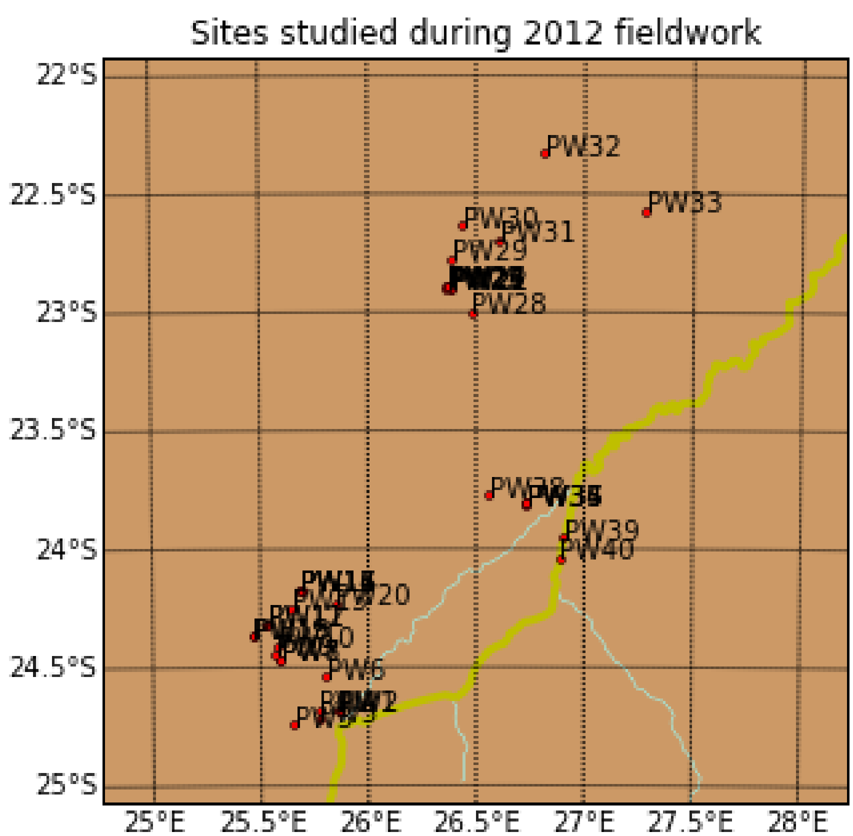
```
In [3]: fig = plt.figure(figsize=(5,5))
    m = Basemap(projection='merc',llcrnrlat=-27.5,urcrnrlat=-17,llcrnrlon=19,
                urcrnrlon=29,lat_ts=-25,resolution='i',area_thresh = 0.1)
    m.drawrivers(color='#99ffff')
    m.drawcoastlines()
    m.drawcountries(linewidth=1.5)
    m.drawmapboundary(fill_color='#99ffff')
    m.fillcontinents(color='#cc9966',lake_color='#99ffff')
    parallels = np.arange(-90,90,2.)
    m.drawparallels(parallels,labels=[1,0,0,0],fontsize=10)
    meridians = np.arange(0.,360.,2.)
    m.drawmeridians(meridians,labels=[0,0,0,1],fontsize=10)
    plt.title('Sites studied during 2012 fieldwork')

    site_long=[]
    site_lat=[]
```

```
for n in range(0,len(Site_Locations)):
    site_long.append(Site_Locations['LONG(WGS84)'][n])
    site_lat.append(Site_Locations['LAT(WGS84)'][n])

x,y = m(site_long,site_lat)
m.plot(x, y, 'ro', markersize=5)
plt.show()
```




```
In [4]: fig = plt.figure(figsize=(5,5))
    m = Basemap(projection='aea',lat_0=-23.5,lon_0=26.5,resolution='f',
                width=350000,height=350000)
    m.drawrivers(color='#99ffff')
    m.drawcoastlines()
    m.drawcountries(linewidth=3, color='y')
    m.drawmapboundary(fill_color='#99ffff')
    m.fillcontinents(color='#cc9966',lake_color='#99ffff')
    parallels = np.arange(-90,90,0.5)
```

```
m.drawparallels(parallels,labels=[1,0,0,0],fontsize=10)
meridians = np.arange(0.,360.,0.5)
m.drawmeridians(meridians,labels=[0,0,0,1],fontsize=10)
plt.title('Sites studied during 2012 fieldwork')

x,y = m(site_long,site_lat)
m.plot(x, y, 'ro', markersize=3)

labels = Site_Locations['ID']
for label, xpt, ypt in zip(labels, x, y):
    plt.text(xpt+1000, ypt, label)

plt.savefig('Code_Output/Botswana_detailed_map_2.pdf')

plt.show()
```



3.3 Analysis of demagnetization data from the newly samples sites

Samples from every site underwent alternating field (AF) demagnetization at the Institute for Rock Magnetism. A subset of sampled localities (31 out of 40) were selected to undergo thermal demagnetization at the UC Berkeley Paleomagnetism Lab. Six samples from each of these 31 localities underwent thermal demagnetization. For both the thermal and AF demagnetizations, samples underwent a liquid nitrogen immersion in a very low field environment after the natural remanent magnetization (NRM) was measured with the goal of removing remanence associated with multidomain magnetite which undergoes low-temperature demagnetization when cycled through the isotropic point (~130 K) and the Verwey transition (~120 K).

Principle component analysis was conducted using on the data using the PmagPy software package as implemented in the demag_gui.py program. Least-squares fits to the data are stored in the pmag_specimens.txt files and are imported below.

```
In [5]: PW1_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW1/pmag_specimens.txt
                              sep='\t',header=1)
        PW3_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW3/pmag_specimens.txt',
                              sep='\t',header=1)
        PW4_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW4/pmag_specimens.txt',
                              sep='\t',header=1)
        PW5_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW5/pmag_specimens.txt',
                              sep='\t',header=1)
        PW6_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW6/pmag_specimens.txt',
                              sep='\t',header=1)
        PW7_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW7/pmag_specimens.txt',
                              sep='\t',header=1)
        PW9_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW9/pmag_specimens.txt',
                              sep='\t',header=1)
        PW10_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW10/pmag_specimens.txt',
                               sep='\t',header=1)
        PW11_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW11/pmag_specimens.txt',
                               sep='\t',header=1)
        PW13_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW13/pmag_specimens.txt',
                               sep='\t',header=1)
        PW15_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW15/pmag_specimens.txt',
                               sep='\t',header=1)
        PW18_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW18/pmag_specimens.txt',
                               sep='\t',header=1)
        PW19_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW19/pmag_specimens.txt',
                               sep='\t',header=1)
        PW20_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW20/pmag_specimens.txt',
                               sep='\t',header=1)
        PW22_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW22/pmag_specimens.txt',
                               sep='\t',header=1)
```

```
PW23_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW23/pmag_specimens.txt',
                      sep='\t',header=1)
PW24_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW24/pmag_specimens.txt',
                      sep='\t',header=1)
PW25_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW25/pmag_specimens.txt',
                      sep='\t',header=1)
PW26_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW26/pmag_specimens.txt',
                      sep='\t',header=1)
PW27_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW27/pmag_specimens.txt',
                      sep='\t',header=1)
PW28_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW28/pmag_specimens.txt',
                      sep='\t',header=1)
PW29_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW29/pmag_specimens.txt',
                      sep='\t',header=1)
PW30_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW30/pmag_specimens.txt',
                      sep='\t',header=1)
PW31_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW31/pmag_specimens.txt',
                      sep='\t',header=1)
PW32_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW32/pmag_specimens.txt',
                      sep='\t',header=1)
PW34_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW34/pmag_specimens.txt',
                      sep='\t',header=1)
PW36_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW36/pmag_specimens.txt',
                      sep='\t',header=1)
PW37_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW37/pmag_specimens.txt',
                      sep='\t',header=1)
PW38_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW38/pmag_specimens.txt',
                      sep='\t',header=1)
PW39_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW39/pmag_specimens.txt',
                      sep='\t',header=1)
PW40_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_HT/PW40/pmag_specimens.txt',
                      sep='\t',header=1)
```

```
In [6]: #slice dataframe to include only tilt corrected fits
    PW1_tc = PW1_data[PW1_data['specimen_tilt_correction'] == 100]
    PW1_tc.reset_index(inplace=True)
    PW3_tc = PW3_data[PW3_data['specimen_tilt_correction'] == 100]
    PW3_tc.reset_index(inplace=True)
    PW4_tc = PW4_data[PW4_data['specimen_tilt_correction'] == 100]
    PW4_tc.reset_index(inplace=True)
    PW5_tc = PW5_data[PW5_data['specimen_tilt_correction'] == 100]
    PW5_tc.reset_index(inplace=True)
    PW6_tc = PW6_data[PW6_data['specimen_tilt_correction'] == 100]
    PW6_tc.reset_index(inplace=True)
    PW7_tc = PW7_data[PW7_data['specimen_tilt_correction'] == 100]
```

```
PW7_tc.reset_index(inplace=True)
PW9_tc = PW9_data[PW9_data['specimen_tilt_correction'] == 100]
PW9_tc.reset_index(inplace=True)
PW10_tc = PW10_data[PW10_data['specimen_tilt_correction'] == 100]
PW10_tc.reset_index(inplace=True)
PW11_tc = PW11_data[PW11_data['specimen_tilt_correction'] == 100]
PW11_tc.reset_index(inplace=True)
PW13_tc = PW13_data[PW13_data['specimen_tilt_correction'] == 100]
PW13_tc.reset_index(inplace=True)
PW15_tc = PW15_data[PW15_data['specimen_tilt_correction'] == 100]
PW15_tc.reset_index(inplace=True)
PW18_tc = PW18_data[PW18_data['specimen_tilt_correction'] == 100]
PW18_tc.reset_index(inplace=True)
PW19_tc = PW19_data[PW19_data['specimen_tilt_correction'] == 100]
PW19_tc.reset_index(inplace=True)
PW20_tc = PW20_data[PW20_data['specimen_tilt_correction'] == 100]
PW20_tc.reset_index(inplace=True)
PW22_tc = PW22_data[PW22_data['specimen_tilt_correction'] == 100]
PW22_tc.reset_index(inplace=True)
PW23_tc = PW23_data[PW23_data['specimen_tilt_correction'] == 100]
PW23_tc.reset_index(inplace=True)
PW24_tc = PW24_data[PW24_data['specimen_tilt_correction'] == 100]
PW24_tc.reset_index(inplace=True)
PW25_tc = PW25_data[PW25_data['specimen_tilt_correction'] == 100]
PW25_tc.reset_index(inplace=True)
PW26_tc = PW26_data[PW26_data['specimen_tilt_correction'] == 100]
PW26_tc.reset_index(inplace=True)
PW27_tc = PW27_data[PW27_data['specimen_tilt_correction'] == 100]
PW27_tc.reset_index(inplace=True)
PW28_tc = PW28_data[PW28_data['specimen_tilt_correction'] == 100]
PW28_tc.reset_index(inplace=True)
PW29_tc = PW29_data[PW29_data['specimen_tilt_correction'] == 100]
PW29_tc.reset_index(inplace=True)
PW30_tc = PW30_data[PW30_data['specimen_tilt_correction'] == 100]
PW30_tc.reset_index(inplace=True)
PW31_tc = PW31_data[PW31_data['specimen_tilt_correction'] == 100]
PW31_tc.reset_index(inplace=True)
PW32_tc = PW32_data[PW32_data['specimen_tilt_correction'] == 100]
PW32_tc.reset_index(inplace=True)
PW34_tc = PW34_data[PW34_data['specimen_tilt_correction'] == 100]
PW34_tc.reset_index(inplace=True)
PW36_tc = PW36_data[PW36_data['specimen_tilt_correction'] == 100]
PW36_tc.reset_index(inplace=True)
PW37_tc = PW37_data[PW37_data['specimen_tilt_correction'] == 100]
PW37_tc.reset_index(inplace=True)
```

```
PW38_tc = PW38_data[PW38_data['specimen_tilt_correction'] == 100]
PW38_tc.reset_index(inplace=True)
PW39_tc = PW39_data[PW39_data['specimen_tilt_correction'] == 100]
PW39_tc.reset_index(inplace=True)
PW40_tc = PW40_data[PW40_data['specimen_tilt_correction'] == 100]
PW40_tc.reset_index(inplace=True)
```

Below is the table with all tilt-correction parameters in dip direction and dip with a short explanation of what orientation is being used. The values are 0 if no tilt-correction was applied. Also shown is the declination correction used for each individual site's bedding. This correction was only used for bedding measurements, as sun compass data were used for orienting paleomagnetic samples. Declination corrections were calculated using IGRF11 and were determined using individual locality coordinates.

```
In [7]: PW_tilt_corrections=pd.read_csv('../Data/Field_Data/Botswana_tilt_corr.csv',header=0)
        PW_tilt_corrections
```

Out[7]:

	Site_ID	Dip_dir	Dip	Dec_cor	Dip_dir_cor	Explanation
0	PW1	240	6	-15.40	224.60	orientation of sill
1	PW2	240	6	-15.40	224.60	orientation of sill
2	PW3	346	5	-15.44	330.56	bedding of Waterberg sandstone that sits atop ...
3	PW4	52	5	-15.38	36.62	bedding of Waterberg sandstone
4	PW5	0	0	-15.44	0.00	no availible orientation information
5	PW6	0	0	-15.20	0.00	no availible orientation information
6	PW7	0	0	-15.06	0.00	appears to be planar-tabular and sub-horizontal
7	PW8	0	0	-15.06	0.00	appears to be planar-tabular and sub-horizontal
8	PW9	0	0	-15.01	0.00	lack orientation information, sill is overall ...
9	PW10	345	7	-14.97	330.03	orientation of sub-horizontal planes that are ...
10	PW11	296	10	-14.84	281.16	bedding of Waterberg sandstone measured near b...
11	PW12	296	10	-14.84	281.16	bedding of Waterberg sandstone measured near b...
12	PW13	0	0	-14.77	0.00	sandstone and contact have low dips (e.g. 3_) ...
13	PW14	275	4	-14.68	260.32	sub-horizontal planes perpendicular to columna...
14	PW15	275	4	-14.68	260.32	sub-horizontal planes perpendicular to columna...
15	PW18	305	4	-14.90	290.10	sill upper contact with Waterber ss
16	PW19	305	4	-14.90	290.10	sill upper contact with Waterber ss, columnar ...
17	PW20	2	3	-14.79	12.79	nearby coarse ss and pebble cgl
18	PW21	276	10	-13.15	262.85	Mosolotsane average of Palapye siltstone/vfn ss
19	PW22	276	10	-13.15	262.85	Mosolotsane average of Palapye siltstone/vfn ss
20	PW23	276	10	-13.14	262.86	Mosolotsane average of Palapye siltstone/vfn ss
21	PW24	276	10	-13.12	262.88	Mosolotsane average of Palapye siltstone/vfn ss
22	PW25	276	10	-13.14	262.86	Mosolotsane average of Palapye siltstone/vfn ss
23	PW26	276	10	-13.12	262.88	Mosolotsane average of Palapye siltstone/vfn ss
24	PW27	276	10	-13.12	262.88	Mosolotsane average of Palapye siltstone/vfn ss
25	PW28	0	0	-13.30	0.00	appears to be planar-tabular and sub-horizontal
26	PW29	283	10	-12.98	270.02	nearest in situ Palapye med ss
27	PW30	0	0	-12.81	0.00	vertical joints suggest no significant tilt of...
28	PW31	103	4	-12.94	90.06	quartzite below sill yields tilt correction
29	PW32	0	0	-12.51	0.00	underlying Palapye vfn ss/siltstone is "effect...
30	PW33	325	9	-12.93	312.07	tilt correction from nearby Palapye siltstone ...
31	PW34	4	8	-14.44	10.44	average of many measurements of nearby Waterbe...
32	PW35	4	8	-14.44	10.44	average of many measurements of nearby Waterbe...
33	PW36	4	8	-14.43	10.43	average of many measurements of nearby Waterbe...
34	PW37	4	8	-14.43	10.43	average of many measurements of nearby Waterbe...
35	PW38	0	0	-14.34	0.00	no sedimentary rock outcrops nearby, sill appe...
36	PW39	0	0	-14.67	0.00	no sedimentary rock outcrops nearby, sill appe...
37	PW40	271	12	-14.79	256.21	average of many measurements of nearby Waterbe...

3.4 Data Reduction and Tabulation of Botswana sites

What follows is a site by site analysis of the sample directions determined through the least-square fits. When interpretable thermal and AF demagnetization data were both developed from specimens of the same sample, the thermal data are preferentially used.

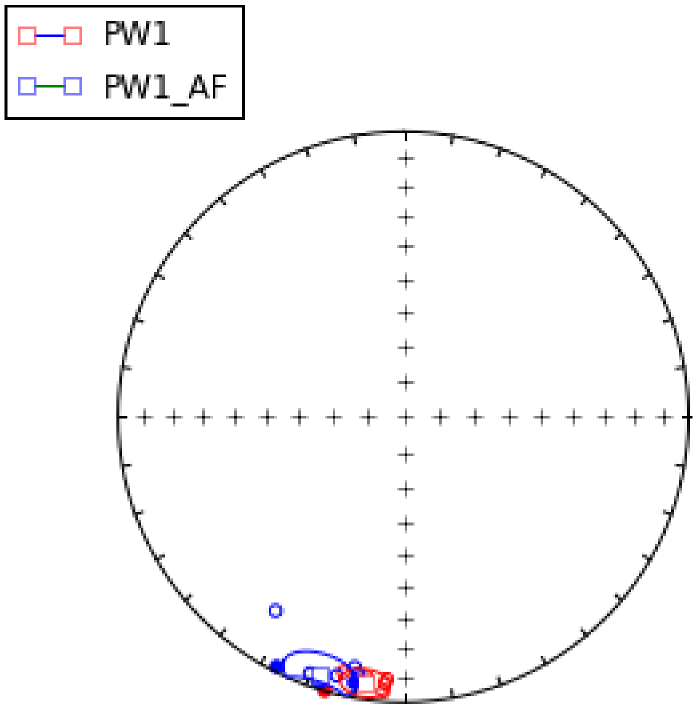
3.4.1 Kgale Peak Sill - PW1, PW2, JP1, JP2, JP3, and J_M9

The behavior of JP1/2/3 (prior data) lacks consistency and these data are unlike PW1, which was very well behaved. 'Site 9' from Jones and McElhinny (1966) is consistent with our results (S-seeking direction); in contrast to the north-seeking direction reported by Pancake (2001).

PW1 All samples from PW1 are well-behaved and consistent.

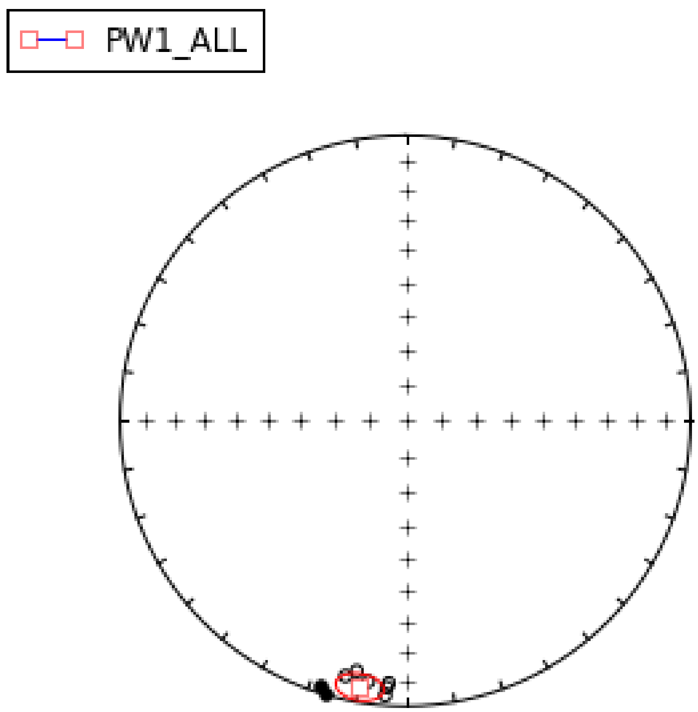
```
In [8]: #Import AF data and plot on same equal-area plot
    PW1_AFdata=pd.read_csv('../Data/Botswana_AF/PW1/pmag_specimens.txt',
                           sep='\t',header=1)
    PW1_AFtc = PW1_AFdata[PW1_AFdata['specimen_tilt_correction'] == 100]
    PW1_AFtc.reset_index(inplace=True)

    PW1_AFtc_dir=[]
    PW1_tc_dir=[]
    for n in range(len(PW1_tc)):
        Dec,Inc=PW1_tc['specimen_dec'][n],PW1_tc['specimen_inc'][n]
        PW1_tc_dir.append([Dec,Inc,1.])
    PW1_tc_mean=pmag.fisher_mean(PW1_tc_dir)
    for n in range(len(PW1_AFtc)):
        Dec,Inc=PW1_AFtc['specimen_dec'][n],PW1_AFtc['specimen_inc'][n]
        PW1_AFtc_dir.append([Dec,Inc,1.])
    PW1_AFtc_mean=pmag.fisher_mean(PW1_AFtc_dir)

    fignum = 1
    plt.figure(num=fignum,figsize=(5,5))
    IPmag.iplotNET(1)
    IPmag.iplotDI(PW1_tc_dir,color='red')
    IPmag.iplotDImean(PW1_tc_mean['dec'],PW1_tc_mean['inc'],
                      PW1_tc_mean["alpha95"],color='r',marker='s',
                      label='PW1')
    IPmag.iplotDI(PW1_AFtc_dir,color='blue')
    IPmag.iplotDImean(PW1_AFtc_mean['dec'],PW1_AFtc_mean['inc'],
                      PW1_AFtc_mean["alpha95"],color='b',marker='s',
                      label='PW1_AF')
    plt.show()
```

```
/Users/Laurentia/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-packages/matplotlib/axe
warnings.warn("No labelled objects found. "
```



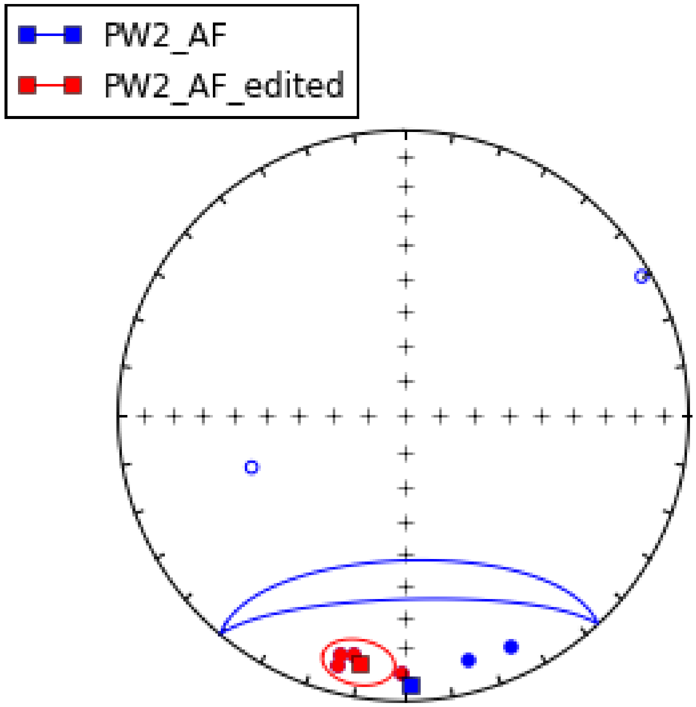

```
In [9]: #Combine AF and thermal data for single mean for this site.
        #We favor THERMAL data points, so if there are duplicate vectors...
        #...for the same sample, the AF vector is dropped
        PW1_ALL = PW1_AFtc
        PW1_ALL = PW1_ALL.append(PW1_tc)
        PW1_ALL.reset_index(drop=True, inplace=True)
        PW1_ALL = PW1_ALL.drop(1)
        PW1_ALL = PW1_ALL.drop(2)
        PW1_ALL = PW1_ALL.drop(3)
        PW1_ALL = PW1_ALL.drop(4)
        PW1_ALL = PW1_ALL.drop(5)
        PW1_ALL = PW1_ALL.drop(6)
        PW1_ALL.reset_index(drop=True, inplace=True)
        PW1_ALL_dir=[]
        for n in range(len(PW1_ALL)):
```

```
    Dec,Inc=PW1_ALL['specimen_dec'][n],PW1_ALL['specimen_inc'][n]
    PW1_ALL_dir.append([Dec,Inc,1.])
PW1_ALL_mean=pmag.fisher_mean(PW1_ALL_dir)
fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW1_ALL_dir,color='k')
IPmag.iplotDImean(PW1_ALL_mean['dec'],PW1_ALL_mean['inc'],
                 PW1_ALL_mean["alpha95"],color='r',marker='s',
                 label='PW1_ALL')
plt.show()
```



PW2 These samples are from the upper chilled margin of the same Kgale Peak sill as PW1. Only AF data were developed for this site. Four samples are very consistent with each other with slightly steeper than results from PW1.

```
In [10]: PW2_AFdata=pd.read_csv('../Data/Botswana_AF/PW2/pmag_specimens.txt',
                               sep='\t',header=1)
         PW2_AFtc = PW2_AFdata[PW2_AFdata['specimen_tilt_correction'] == 100]
         PW2_AFtc.reset_index(drop=True, inplace=True)

         PW2_AFtc_dir=[]
         for n in range(len(PW2_AFtc)):
             Dec,Inc=PW2_AFtc['specimen_dec'][n],PW2_AFtc['specimen_inc'][n]
             PW2_AFtc_dir.append([Dec,Inc,1.])
         PW2_AFtc_mean=pmag.fisher_mean(PW2_AFtc_dir)

         #Drop samples already measured with thermal demag
         PW2_AFtc_edit = PW2_AFtc
         PW2_AFtc_edit = PW2_AFtc_edit.drop(0)
         PW2_AFtc_edit = PW2_AFtc_edit.drop(1)
         PW2_AFtc_edit = PW2_AFtc_edit.drop(4)
         PW2_AFtc_edit = PW2_AFtc_edit.drop(5)
         PW2_AFtc_edit.reset_index(drop=True, inplace=True)
         PW2_AFtc_edit_dir=[]
         for n in range(len(PW2_AFtc_edit)):
             Dec,Inc=PW2_AFtc_edit['specimen_dec'][n],PW2_AFtc_edit['specimen_inc'][n]
             PW2_AFtc_edit_dir.append([Dec,Inc,1.])
         PW2_AFtc_edit_mean=pmag.fisher_mean(PW2_AFtc_edit_dir)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(PW2_AFtc_dir,color='blue')
         IPmag.iplotDImean(PW2_AFtc_mean['dec'],PW2_AFtc_mean['inc'],
                           PW2_AFtc_mean["alpha95"],color='b',marker='s',
                           label='PW2_AF')
         IPmag.iplotDI(PW2_AFtc_edit_dir,color='red')
         IPmag.iplotDImean(PW2_AFtc_edit_mean['dec'],PW2_AFtc_edit_mean['inc'],
                           PW2_AFtc_edit_mean["alpha95"],color='r',marker='s',
                           label='PW2_AF_edited')
```

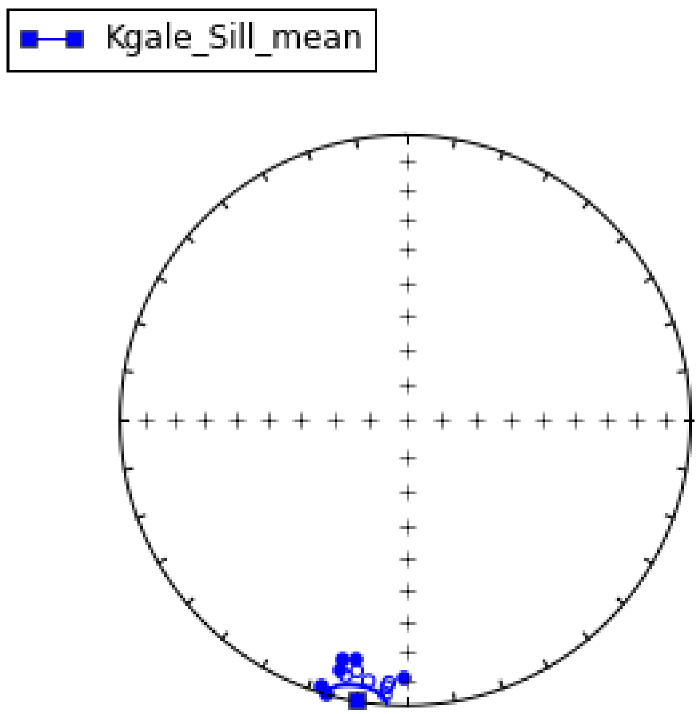


JP(1,2,3) Both PW1 and PW2 are from the same intrusion as Pancake's JP1, JP2, and JP3 data. Only site JP1 yielded stable results, a mean with an α_{95} of 19.1 and has a northerly declination. Our thermal and AF results seem to be more stable and consistent, especially considering Pancake's other sites from the same sill PW2 and PW3 did not yield stable results. Notably, both our data and that from Jones and McElhinny (1966) indicate a south-seeking direction for the sill.

Combined Kgale sill mean - PW1_ALL, PW2_AF

```
In [11]: Kgale_Sill=[]
     Kgale_Sill = PW2_AFtc_edit_dir + PW1_ALL_dir
     Kgale_Sill_mean=pmag.fisher_mean(Kgale_Sill)

     fignum = 1
```

```
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(Kgale_Sill,color='blue')
IPmag.iplotDImean(Kgale_Sill_mean['dec'],Kgale_Sill_mean['inc'],
                  Kgale_Sill_mean["alpha95"],color='b',marker='s',
                  label='Kgale_Sill_mean')
```




```
In [12]: Intrusion_mean_directions = pd.DataFrame(columns=['Intrusion_name',
                                         'sites_used','site_lat','site_long',
                                         'n','dec_tc','inc_tc','a_95','k',
                                         'date','date_error','dip_direction','dip'])
     Intrusion_mean_directions.loc['Kgale_Peak_Sill']=pd.Series({'Intrusion_name':
                                    'Kgale_Peak_Sill',
                                    'sites_used':'PW1_ALL and PW2_AF',
                                    'site_lat':Site_Locations['LAT(WGS84)'][0],
```

```
                                              'site_long':Site_Locations['LONG(WGS84)'][0],
                                              'n':int(Kgale_Sill_mean['n']),
                                              'dec_tc':round(Kgale_Sill_mean['dec'],1),
                                              'inc_tc':round(Kgale_Sill_mean['inc'],1),
                                              'a_95':round(Kgale_Sill_mean['alpha95'],1),
                                              'k':round(Kgale_Sill_mean['k'],1),
                                              'date':'1108.0',
                                              'date_error':'0.9',
                                              'dip_direction':224.6,
                                              'dip':6})
     Intrusion_mean_directions
```

Out[12]:

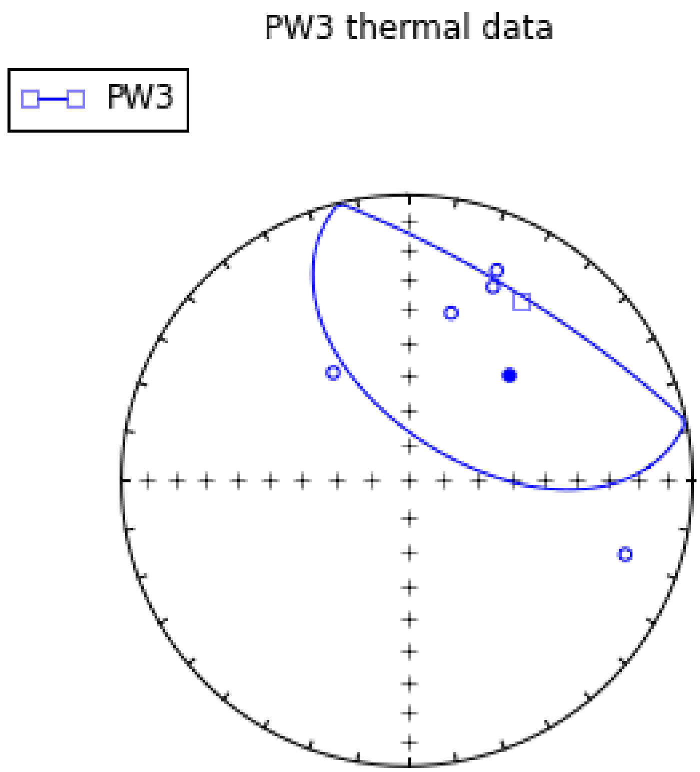
	Intrusion_name	sites_used	site_lat	site_long	n	dec_tc	inc_tc	a_95	k	date	date_error	dip_directi
Kgale_Peak_Sill	Kgale_Peak_Sill	PW1_ALL and PW2_AF	-24.68781	25.86215	12	189.9	0.4	6.3	48.8	1108.0	0.9	22

We will continue to add to the table above as more VGPs are calculated. The full table will be shown at the end of the data analysis.

3.4.2 Manyana Sill - PW3

Thermal data Some samples from PW3 are well-behaved and consistent, but three give scattered directions.

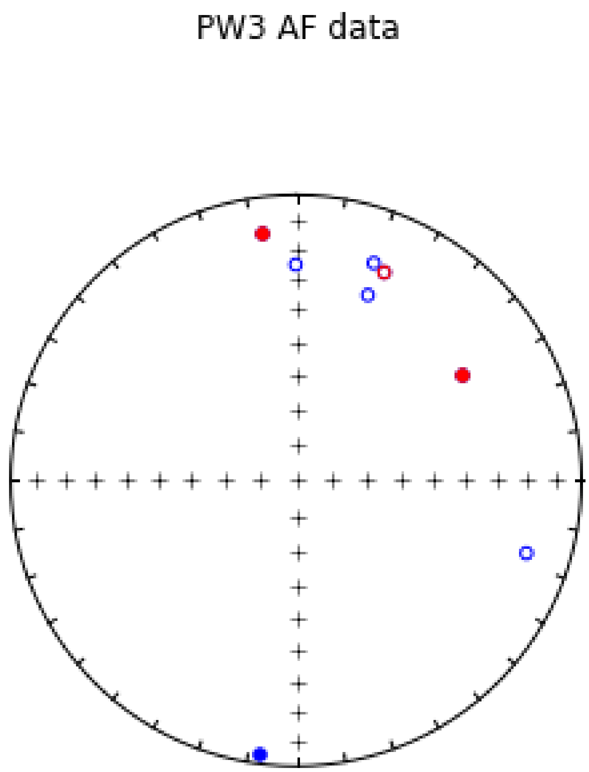
```
In [13]: PW3_tc_dir=[]
         for n in range(len(PW3_tc)):
             Dec,Inc=PW3_tc['specimen_dec'][n],PW3_tc['specimen_inc'][n]
             PW3_tc_dir.append([Dec,Inc,1.])
         PW3_tc_mean=pmag.fisher_mean(PW3_tc_dir)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(PW3_tc_dir,color='b')
         IPmag.iplotDImean(PW3_tc_mean['dec'],PW3_tc_mean['inc'],
                           PW3_tc_mean["alpha95"],color='b',marker='s',label='PW3')
         plt.title('PW3 thermal data')
         plt.show()
```



AF data

```
In [14]: PW3_AFdata=pd.read_csv('../Data/Botswana_AF/PW3/pmag_specimens.txt',
                                 sep='\t',header=1)
         PW3_AFtc = PW3_AFdata[PW3_AFdata['specimen_tilt_correction'] == 100]
         PW3_AFtc.reset_index(drop=True, inplace=True)

         PW3_AFtc_dir=[]
         for n in range(len(PW3_AFtc)):
             Dec,Inc=PW3_AFtc['specimen_dec'][n],PW3_AFtc['specimen_inc'][n]
             PW3_AFtc_dir.append([Dec,Inc,1.])
         PW3_AFtc_mean=pmag.fisher_mean(PW3_AFtc_dir)

         #Drop pts from mean that are AF specimens from same samples as Thermal demag..
```

```
PW3_AFtc_edit = PW3_AFtc
PW3_AFtc_edit = PW3_AFtc_edit.drop(0)
PW3_AFtc_edit = PW3_AFtc_edit.drop(1)
PW3_AFtc_edit = PW3_AFtc_edit.drop(3)
PW3_AFtc_edit = PW3_AFtc_edit.drop(4)
PW3_AFtc_edit = PW3_AFtc_edit.drop(5)
PW3_AFtc_edit.reset_index(drop=True, inplace=True)
PW3_AFtc_edit_dir=[]
for n in range(len(PW3_AFtc_edit)):
    Dec,Inc=PW3_AFtc_edit['specimen_dec'][n],PW3_AFtc_edit['specimen_inc'][n]
    PW3_AFtc_edit_dir.append([Dec,Inc,1.])
PW3_AFtc_edit_mean=pmag.fisher_mean(PW3_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW3_AFtc_dir,color='blue',label='redundant AF data')
IPmag.iplotDI(PW3_AFtc_edit_dir,color='red',label='AF data from new samples')
plt.title('PW3 AF data')
plt.show()
```

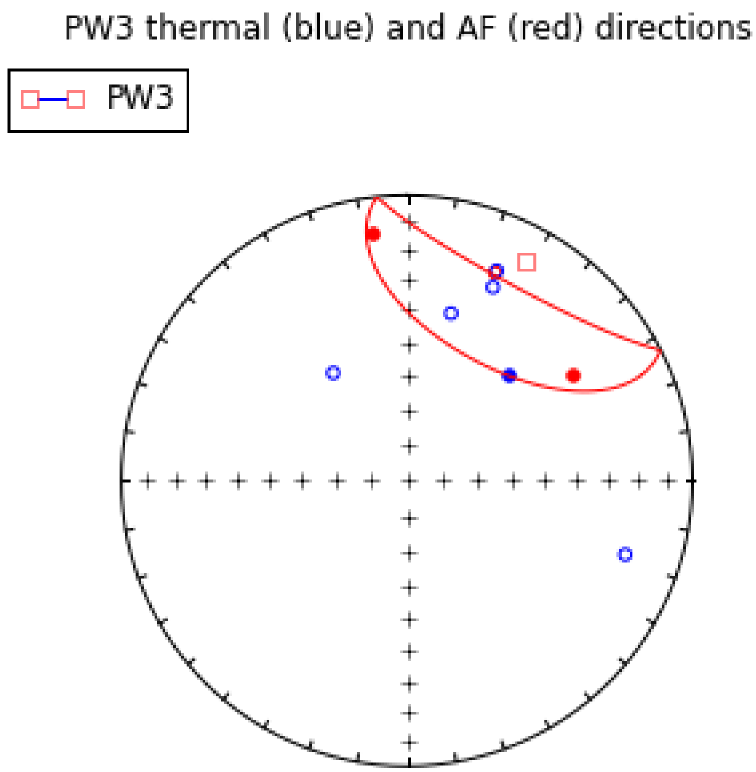


Combined Manyana Sill Mean - PW3_ALL There is a grouping of samples plotting NNE and up. However we determined that the results too scattered for us to consider that we have a well-determined direction and the data are not added to the compilation.

```
In [15]: Manyana_Sill=[]
         Manyana_Sill = PW3_AFtc_edit_dir + PW3_tc_dir
         Manyana_Sill_mean=pmag.fisher_mean(Manyana_Sill)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(PW3_AFtc_edit_dir,color='red')
         IPmag.iplotDI(PW3_tc_dir,color='blue')
         IPmag.iplotDImean(Manyana_Sill_mean['dec'],Manyana_Sill_mean['inc'],
```

```
                    Manyana_Sill_mean["alpha95"],color='r',marker='s',
                    label='PW3')
plt.title('PW3 thermal (blue) and AF (red) directions')
Manyana_Sill_mean
```

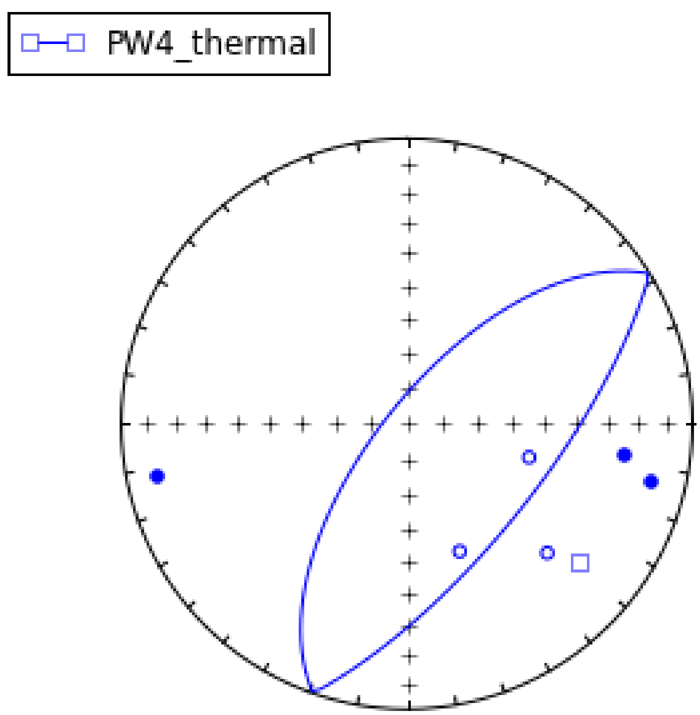
```
Out[15]: {'alpha95': 36.583276959714155,
      'csd': 47.254004096267487,
      'dec': 28.364455683342069,
      'inc': -13.383785314736791,
      'k': 2.9382774935077776,
      'n': 9,
      'r': 6.2773163808808841}
```



3.4.3 Ranku Hill Sill - PW4

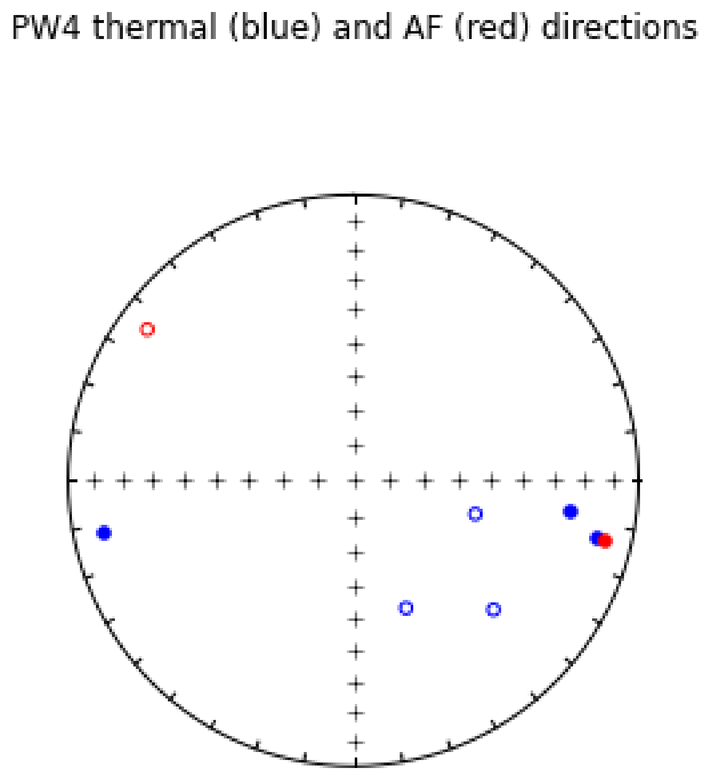
Thermal demag

```
In [16]: PW4_tc_dir=[]
    for n in range(len(PW4_tc)):
        Dec,Inc=PW4_tc['specimen_dec'][n],PW4_tc['specimen_inc'][n]
        PW4_tc_dir.append([Dec,Inc,1.])
    PW4_tc_mean=pmag.fisher_mean(PW4_tc_dir)

    fignum = 1
    plt.figure(num=fignum,figsize=(5,5))
    IPmag.iplotNET(1)
    IPmag.iplotDI(PW4_tc_dir,color='blue')
    IPmag.iplotDImean(PW4_tc_mean['dec'],PW4_tc_mean['inc'],
                      PW4_tc_mean["alpha95"],color='b',marker='s',
                      label='PW4_thermal')
```



AF demag

```
In [17]: PW4_AFdata=pd.read_csv('../Data/Botswana_AF/PW4/pmag_specimens.txt',
                                sep='\t',header=1)
         PW4_AFtc = PW4_AFdata[PW4_AFdata['specimen_tilt_correction'] == 100]
         PW4_AFtc.reset_index(drop=True, inplace=True)

         PW4_AFtc_dir=[]
         for n in range(len(PW4_AFtc)):
             Dec,Inc=PW4_AFtc['specimen_dec'][n],PW4_AFtc['specimen_inc'][n]
             PW4_AFtc_dir.append([Dec,Inc,1.])
         PW4_AFtc_mean=pmag.fisher_mean(PW4_AFtc_dir)

         #Drop redundant specimens from same sample out of AF group
         PW4_AFtc_edit = PW4_AFtc
         PW4_AFtc_edit = PW4_AFtc_edit.drop(1)
         PW4_AFtc_edit = PW4_AFtc_edit.drop(2)
         PW4_AFtc_edit = PW4_AFtc_edit.drop(3)
         PW4_AFtc_edit = PW4_AFtc_edit.drop(4)
         PW4_AFtc_edit = PW4_AFtc_edit.drop(5)
         PW4_AFtc_edit = PW4_AFtc_edit.drop(6)
         PW4_AFtc_edit.reset_index(drop=True, inplace=True)
         PW4_AFtc_edit_dir=[]
         for n in range(len(PW4_AFtc_edit)):
             Dec,Inc=PW4_AFtc_edit['specimen_dec'][n],PW4_AFtc_edit['specimen_inc'][n]
             PW4_AFtc_edit_dir.append([Dec,Inc,1.])
         PW4_AFtc_edit_mean=pmag.fisher_mean(PW4_AFtc_edit_dir)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(PW4_tc_dir,color='blue')
         IPmag.iplotDI(PW4_AFtc_edit_dir,color='red')
         plt.title('PW4 thermal (blue) and AF (red) directions')
         plt.show()
```



This site does not yield a consistent mean and is not included in the compilation dataset.

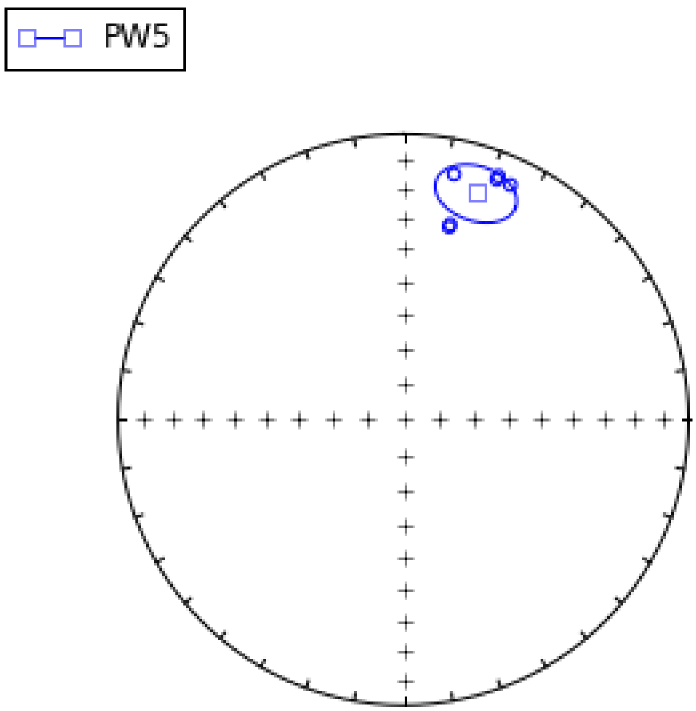
3.4.4 Rasemong North intrusion - PW5

There is good agreement among all samples yielding a north-seeking direction that we interpret as an Umkondo LIP direction.

Thermal demag

```
In [18]: PW5_tc_dir=[]
         for n in range(len(PW5_tc)):
             Dec,Inc=PW5_tc['specimen_dec'][n],PW5_tc['specimen_inc'][n]
             PW5_tc_dir.append([Dec,Inc,1.])
         PW5_tc_mean=pmag.fisher_mean(PW5_tc_dir)
```

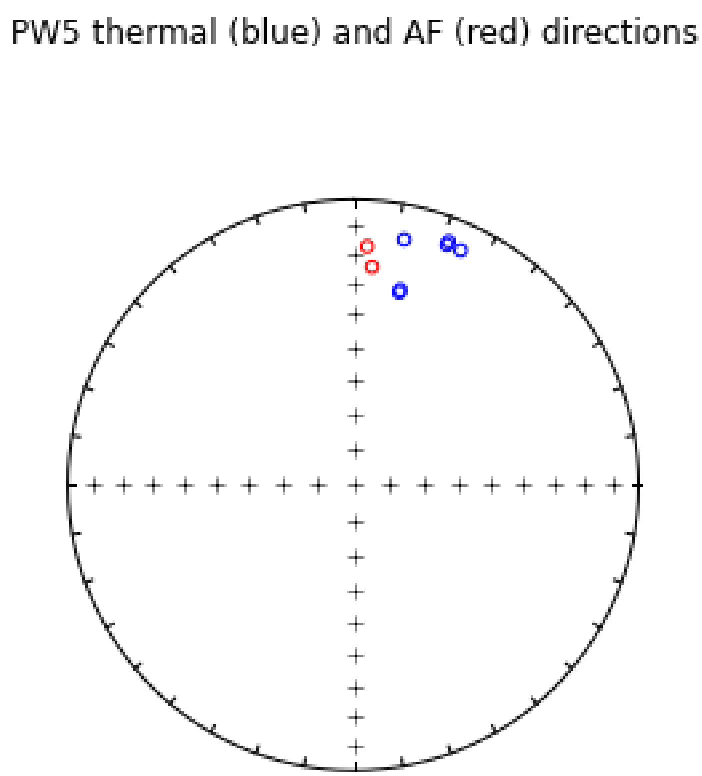
```
fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW5_tc_dir,color='b')
IPmag.iplotDImean(PW5_tc_mean['dec'],PW5_tc_mean['inc'],
                  PW5_tc_mean["alpha95"],color='b',marker='s',
                  label='PW5')
```



AF demag

```
In [19]: PW5_AFdata=pd.read_csv('../Data/Botswana_AF/PW5/pmag_specimens.txt',
                               sep='\t',header=1)
     PW5_AFtc = PW5_AFdata[PW5_AFdata['specimen_tilt_correction'] == 100]
```

```
PW5_AFtc.reset_index(drop=True, inplace=True)

PW5_AFtc_dir=[]
for n in range(len(PW5_AFtc)):
    Dec,Inc=PW5_AFtc['specimen_dec'][n],PW5_AFtc['specimen_inc'][n]
    PW5_AFtc_dir.append([Dec,Inc,1.])
PW5_AFtc_mean=pmag.fisher_mean(PW5_AFtc_dir)

#Drop redundant specimens from same sample out of AF group
PW5_AFtc_edit = PW5_AFtc
PW5_AFtc_edit = PW5_AFtc_edit.drop(0)
PW5_AFtc_edit = PW5_AFtc_edit.drop(1)
PW5_AFtc_edit = PW5_AFtc_edit.drop(2)
PW5_AFtc_edit = PW5_AFtc_edit.drop(4)
PW5_AFtc_edit = PW5_AFtc_edit.drop(5)
PW5_AFtc_edit = PW5_AFtc_edit.drop(6)
PW5_AFtc_edit.reset_index(drop=True, inplace=True)
PW5_AFtc_edit_dir=[]
for n in range(len(PW5_AFtc_edit)):
    Dec,Inc=PW5_AFtc_edit['specimen_dec'][n],PW5_AFtc_edit['specimen_inc'][n]
    PW5_AFtc_edit_dir.append([Dec,Inc,1.])
PW5_AFtc_edit_mean=pmag.fisher_mean(PW5_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW5_tc_dir,color='blue')
IPmag.iplotDI(PW5_AFtc_edit_dir,color='red')
plt.title('PW5 thermal (blue) and AF (red) directions')
plt.show()
```

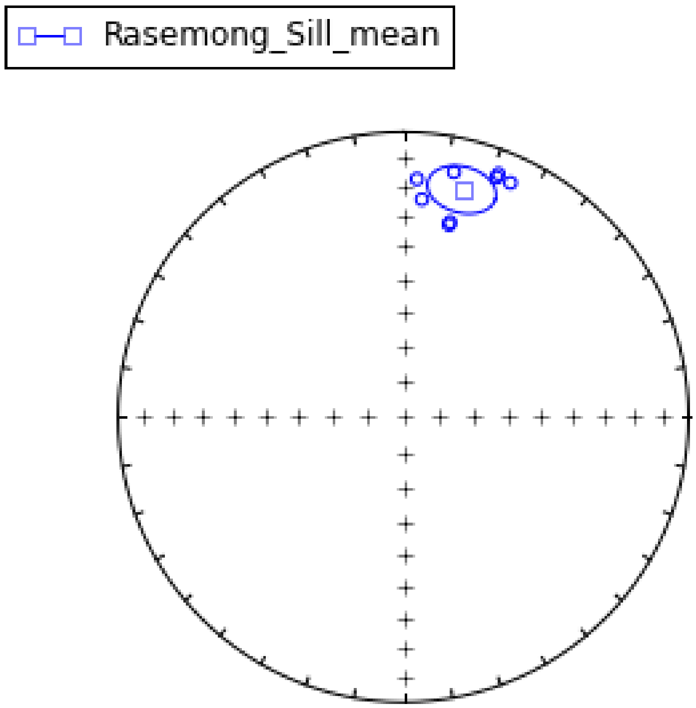


There is very good agreement between all of the thermal and AF results. Two samples with AF results were added to the mean - these samples lacked thermal demagnetization data.

Combined Rasemong Sill Mean - PW5_ALL

```
In [20]: Rasemong_Sill=[]
         Rasemong_Sill = PW5_AFtc_edit_dir + PW5_tc_dir
         Rasemong_Sill_mean=pmag.fisher_mean(Rasemong_Sill)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(Rasemong_Sill,color='blue')
         IPmag.iplotDImean(Rasemong_Sill_mean['dec'],Rasemong_Sill_mean['inc'],
                           Rasemong_Sill_mean["alpha95"],color='b',marker='s',
```

```
label='Rasemong_Sill_mean')
```




```
In [21]: Intrusion_mean_directions.loc['Rasemong_Sill']=pd.Series({'Intrusion_name':
                                        'Rasemong_Sill','sites_used':'PW5_ALL',
                                        'site_lat':Site_Locations['LAT(WGS84)'][4],
                                        'site_long':Site_Locations['LONG(WGS84)'][4],
                                        'n':int(Rasemong_Sill_mean['n']),
                                        'dec_tc':round(Rasemong_Sill_mean['dec'],1),
                                        'inc_tc':round(Rasemong_Sill_mean['inc'],1),
                                        'a_95':round(Rasemong_Sill_mean['alpha95'],1),
                                        'k':round(Rasemong_Sill_mean['k'],1),
                                        'dip_direction':0,
                                        'dip':0})
     Intrusion_mean_directions.ix['Rasemong_Sill']
```

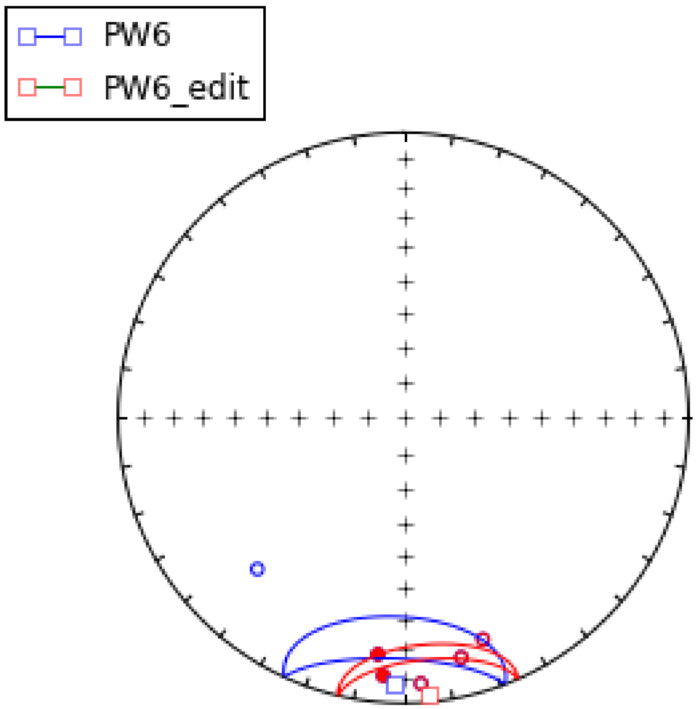
```
Out[21]: Intrusion_name    Rasemong_Sill
         sites_used          PW5_ALL
         site_lat           -24.72715
         site_long            25.7759
         n                          8
         dec_tc                  14.4
         inc_tc                 -18.6
         a_95                     8.1
         k                         48
         date                     NaN
         date_error               NaN
         dip_direction              0
         dip                        0
         Name: Rasemong_Sill, dtype: object
```

3.4.5 Metsemotlhaba River Sill - PW6

Thermal demag

```
In [22]: PW6_tc_dir=[]
         for n in range(len(PW6_tc)):
             Dec,Inc=PW6_tc['specimen_dec'][n],PW6_tc['specimen_inc'][n]
             PW6_tc_dir.append([Dec,Inc,1.])
         PW6_tc_mean=pmag.fisher_mean(PW6_tc_dir)

         #Excluding one outlier point
         PW6_tc_edit = PW6_tc
         PW6_tc_edit = PW6_tc_edit.drop(0)
         PW6_tc_edit.reset_index(inplace=True)

         PW6_tc_edit_dir=[]
         for n in range(len(PW6_tc_edit)):
             Dec,Inc=PW6_tc_edit['specimen_dec'][n],PW6_tc_edit['specimen_inc'][n]
             PW6_tc_edit_dir.append([Dec,Inc,1.])
         PW6_tc_edit_mean=pmag.fisher_mean(PW6_tc_edit_dir)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(PW6_tc_dir,color='b')
         IPmag.iplotDImean(PW6_tc_mean['dec'],PW6_tc_mean['inc'],
                           PW6_tc_mean["alpha95"],color='b',marker='s',
                           label='PW6')
         IPmag.iplotDI(PW6_tc_edit_dir,color='r')
```

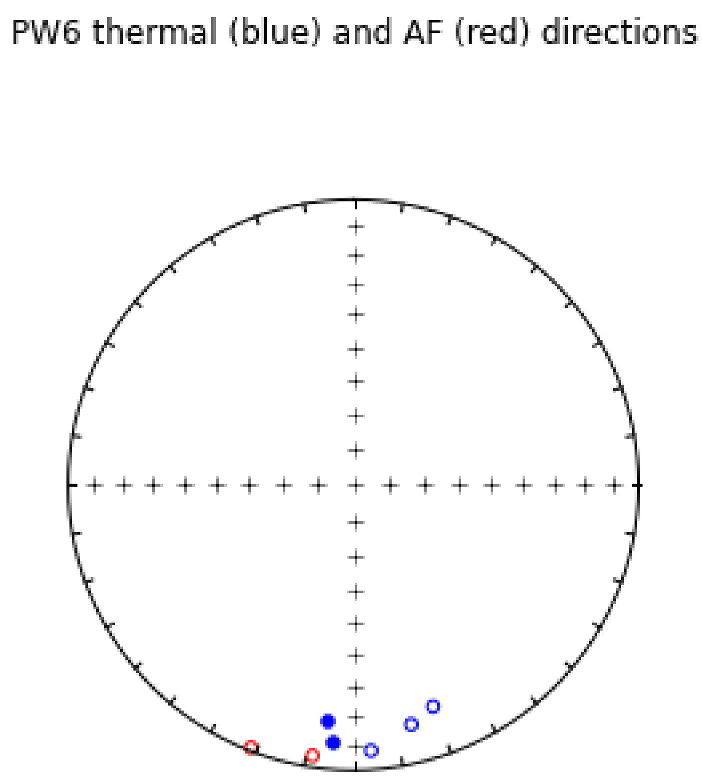
```
IPmag.iplotDImean(PW6_tc_edit_mean['dec'],PW6_tc_edit_mean['inc'],
                  PW6_tc_edit_mean["alpha95"],color='r',marker='s',
                  label='PW6_edit')
```



AF demag

```
In [23]: PW6_AFdata=pd.read_csv('../Data/Botswana_AF/PW6/pmag_specimens.txt',
                                sep='\t',header=1)
         PW6_AFtc = PW6_AFdata[PW6_AFdata['specimen_tilt_correction'] == 100]
         PW6_AFtc.reset_index(drop=True, inplace=True)

         PW6_AFtc_dir=[]
         for n in range(len(PW6_AFtc)):
             Dec,Inc=PW6_AFtc['specimen_dec'][n],PW6_AFtc['specimen_inc'][n]
```

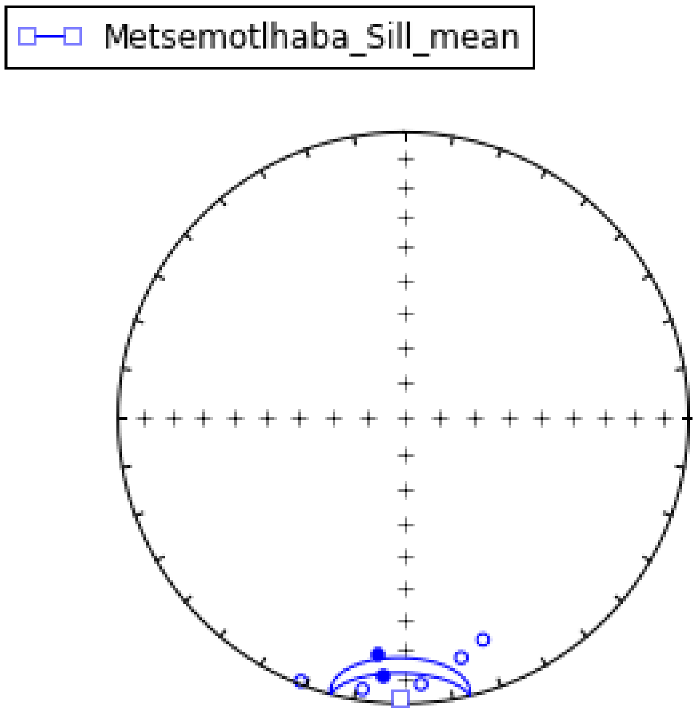
```
    PW6_AFtc_dir.append([Dec,Inc,1.])
PW6_AFtc_mean=pmag.fisher_mean(PW6_AFtc_dir)

#Drop redundant specimens from same sample out of AF group
PW6_AFtc_edit = PW6_AFtc
PW6_AFtc_edit = PW6_AFtc_edit.drop(2)
PW6_AFtc_edit = PW6_AFtc_edit.drop(3)
PW6_AFtc_edit = PW6_AFtc_edit.drop(4)
PW6_AFtc_edit = PW6_AFtc_edit.drop(5)
PW6_AFtc_edit = PW6_AFtc_edit.drop(6)
PW6_AFtc_edit.reset_index(drop=True, inplace=True)
PW6_AFtc_edit_dir=[]
for n in range(len(PW6_AFtc_edit)):
    Dec,Inc=PW6_AFtc_edit['specimen_dec'][n],PW6_AFtc_edit['specimen_inc'][n]
    PW6_AFtc_edit_dir.append([Dec,Inc,1.])
PW6_AFtc_edit_mean=pmag.fisher_mean(PW6_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW6_tc_edit_dir,color='blue')
IPmag.iplotDI(PW6_AFtc_edit_dir,color='red')
plt.title('PW6 thermal (blue) and AF (red) directions')
plt.show()
```



Metsemotlhaba River Sill combined mean - PW6_ALL There is agreement between the thermal and AF results. Two AF samples can be added to the PW6 mean.

```
In [24]: Metsemotlhaba_Sill=[]
    Metsemotlhaba_Sill = PW6_AFtc_edit_dir + PW6_tc_edit_dir
    Metsemotlhaba_Sill_mean=pmag.fisher_mean(Metsemotlhaba_Sill)

    fignum = 1
    plt.figure(num=fignum,figsize=(5,5))
    IPmag.iplotNET(1)
    IPmag.iplotDI(Metsemotlhaba_Sill,color='blue')
    IPmag.iplotDImean(Metsemotlhaba_Sill_mean['dec'],
                      Metsemotlhaba_Sill_mean['inc'],
```

```
Metsemotlhaba_Sill_mean["alpha95"],color='b',
marker='s',label='Metsemotlhaba_Sill_mean')
```




```
In [25]: Intrusion_mean_directions.loc['Metsemotlhaba_River_Sill']=pd.Series({
             'Intrusion_name':'Metsemotlhaba River Sill',
             'sites_used':'PW6_ALL','site_lat':Site_Locations['LAT(WGS84)'][5],
             'site_long':Site_Locations['LONG(WGS84)'][5],
             'n':int(Metsemotlhaba_Sill_mean['n']),
             'dec_tc':round(Metsemotlhaba_Sill_mean['dec'],1),
             'inc_tc':round(Metsemotlhaba_Sill_mean['inc'],1),
             'a_95':round(Metsemotlhaba_Sill_mean['alpha95'],1),
             'k':round(Metsemotlhaba_Sill_mean['k'],1),
             'dip_direction':0,
             'dip':0})
     Intrusion_mean_directions.ix['Metsemotlhaba_River_Sill']
```

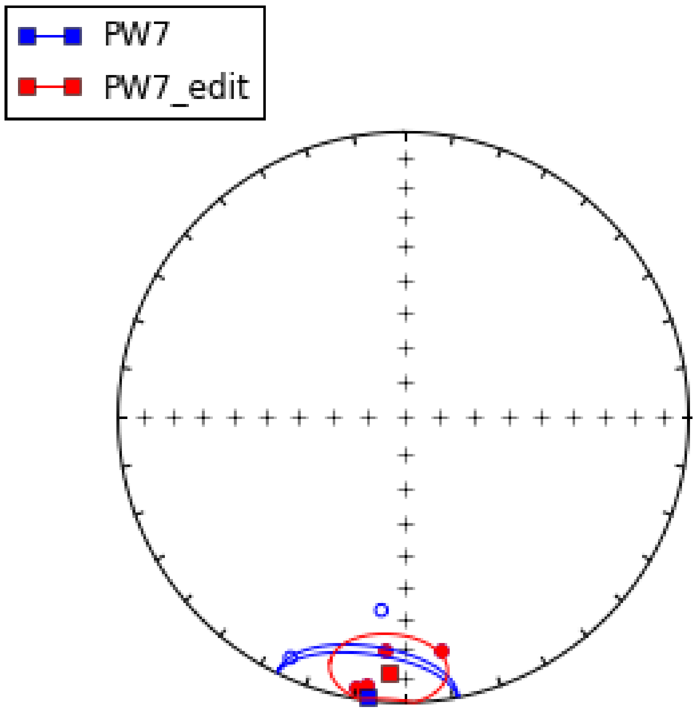
```
Out[25]: Intrusion_name    Metsemotlhaba River Sill
         sites_used                        PW6_ALL
         site_lat                        -24.54694
         site_long                        25.80892
         n                                       7
         dec_tc                              180.6
         inc_tc                               -2.7
         a_95                                 14.4
         k                                    18.5
         date                                  NaN
         date_error                            NaN
         dip_direction                           0
         dip                                     0
         Name: Metsemotlhaba_River_Sill, dtype: object
```

3.4.6 Mabogoapitse Hill Sill - PW7,PW8

Thermal demag (PW7, not PW8)

```
In [26]: PW7_tc_dir=[]
         for n in range(len(PW7_tc)):
             Dec,Inc=PW7_tc['specimen_dec'][n],PW7_tc['specimen_inc'][n],
             PW7_tc_dir.append([Dec,Inc,1.])
         PW7_tc_mean=pmag.fisher_mean(PW7_tc_dir)

         #Excluding one point on the upper hemisphere and sample 10 - replcd w/AF-10:
         PW7_tc_edit = PW7_tc
         PW7_tc_edit = PW7_tc_edit.drop(0)
         PW7_tc_edit = PW7_tc_edit.drop(5)
         PW7_tc_edit.reset_index(inplace=True)

         PW7_tc_edit_dir=[]
         for n in range(len(PW7_tc_edit)):
             Dec,Inc=PW7_tc_edit['specimen_dec'][n],PW7_tc_edit['specimen_inc'][n],
             PW7_tc_edit_dir.append([Dec,Inc,1.])
         PW7_tc_edit_mean=pmag.fisher_mean(PW7_tc_edit_dir)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(PW7_tc_dir,color='b')
         IPmag.iplotDImean(PW7_tc_mean['dec'],PW7_tc_mean['inc'],
                           PW7_tc_mean["alpha95"],color='b',marker='s',label='PW7')
         IPmag.iplotDI(PW7_tc_edit_dir,color='r')
```

```
IPmag.iplotDImean(PW7_tc_edit_mean['dec'],PW7_tc_edit_mean['inc'],
                  PW7_tc_edit_mean["alpha95"],color='r',marker='s',
                  label='PW7_edit')
```

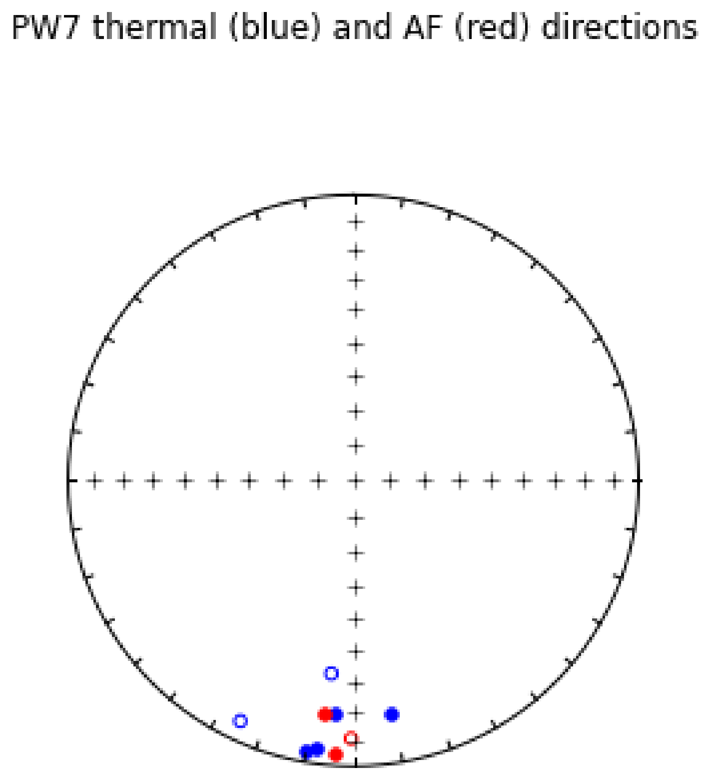


AF demag (PW7 and PW8)

PW7-AF

```
In [27]: PW7_AFdata=pd.read_csv('../Data/Botswana_AF/PW7/pmag_specimens.txt',
                                sep='\t',header=1)
         PW7_AFtc = PW7_AFdata[PW7_AFdata['specimen_tilt_correction'] == 100]
         PW7_AFtc.reset_index(drop=True, inplace=True)

         PW7_AFtc_dir=[]
```

```
for n in range(len(PW7_AFtc)):
    Dec,Inc=PW7_AFtc['specimen_dec'][n],PW7_AFtc['specimen_inc'][n]
    PW7_AFtc_dir.append([Dec,Inc,1.])
PW7_AFtc_mean=pmag.fisher_mean(PW7_AFtc_dir)

#Drop redundant specimens from same sample out of AF group
PW7_AFtc_edit = PW7_AFtc
PW7_AFtc_edit = PW7_AFtc_edit.drop(3)
PW7_AFtc_edit = PW7_AFtc_edit.drop(4)
PW7_AFtc_edit = PW7_AFtc_edit.drop(5)
PW7_AFtc_edit = PW7_AFtc_edit.drop(6)
PW7_AFtc_edit.reset_index(drop=True, inplace=True)
PW7_AFtc_edit_dir=[]
for n in range(len(PW7_AFtc_edit)):
    Dec,Inc=PW7_AFtc_edit['specimen_dec'][n],PW7_AFtc_edit['specimen_inc'][n]
    PW7_AFtc_edit_dir.append([Dec,Inc,1.])
PW7_AFtc_edit_mean=pmag.fisher_mean(PW7_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW7_tc_dir,color='blue')
IPmag.iplotDI(PW7_AFtc_edit_dir,color='red')
plt.title('PW7 thermal (blue) and AF (red) directions')
plt.show()
```

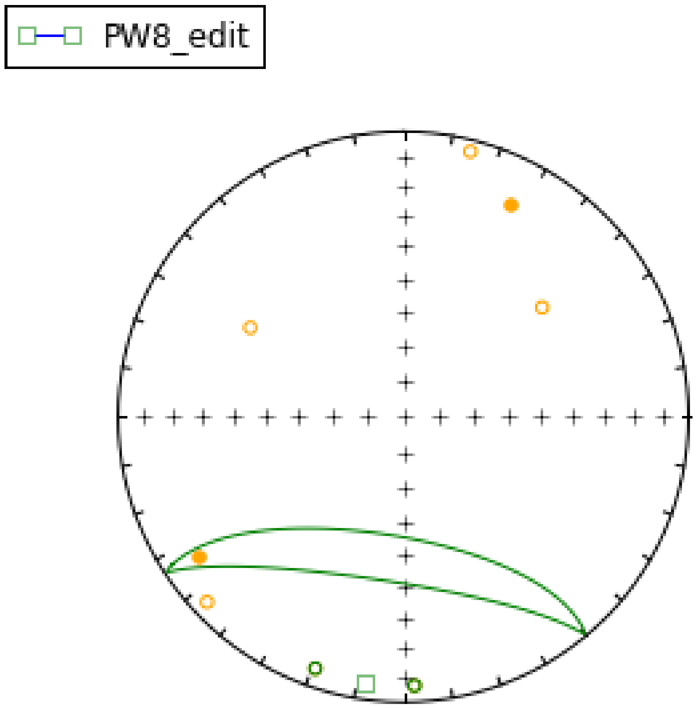


All thermal and AF results share a consistent direction. Above has thermal (blue) and AF (red) specimens. We replace the thermal sample PW7-10(thermal) with PW7-10(AF). Thermal analysis don't include samples PW7-11 and 12 - the AF specimens are included.

PW8-AF

```
In [28]: PW8_AFdata=pd.read_csv('../Data/Botswana_AF/PW8/pmag_specimens.txt',
                                sep='\t',header=1)
         PW8_AFtc = PW8_AFdata[PW8_AFdata['specimen_tilt_correction'] == 100]
         PW8_AFtc.reset_index(drop=True, inplace=True)

         PW8_AFtc_dir=[]
         for n in range(len(PW8_AFtc)):
             Dec,Inc=PW8_AFtc['specimen_dec'][n],PW8_AFtc['specimen_inc'][n]
             PW8_AFtc_dir.append([Dec,Inc,1.])
```

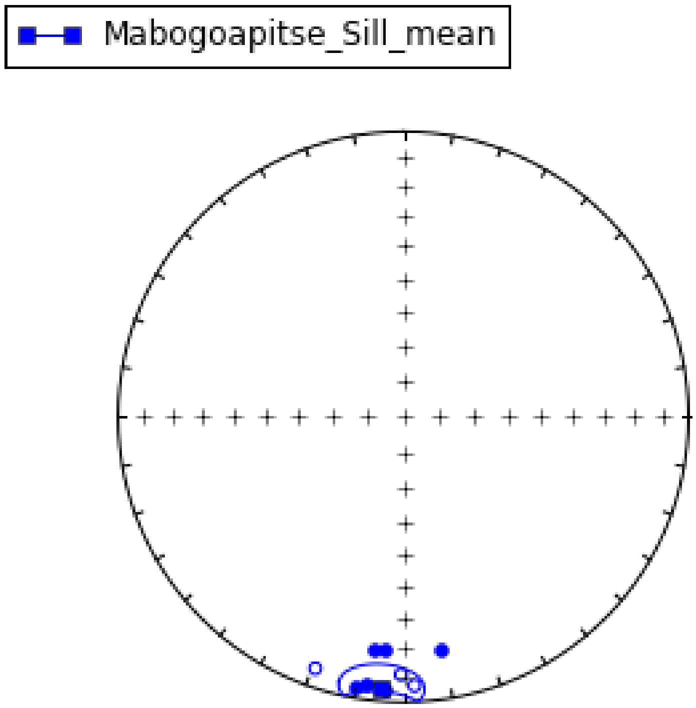
```
PW8_AFtc_mean=pmag.fisher_mean(PW8_AFtc_dir)

#Drop redundant specimens from same sample out of AF group...
#...and two others
PW8_AFtc_edit = PW8_AFtc
PW8_AFtc_edit = PW8_AFtc_edit.drop(2)
PW8_AFtc_edit = PW8_AFtc_edit.drop(3)
PW8_AFtc_edit = PW8_AFtc_edit.drop(4)
PW8_AFtc_edit = PW8_AFtc_edit.drop(5)
PW8_AFtc_edit = PW8_AFtc_edit.drop(0)
PW8_AFtc_edit = PW8_AFtc_edit.drop(1)
PW8_AFtc_edit.reset_index(drop=True, inplace=True)
PW8_AFtc_edit_dir=[]
for n in range(len(PW8_AFtc_edit)):
    Dec,Inc=PW8_AFtc_edit['specimen_dec'][n],PW8_AFtc_edit['specimen_inc'][n]
    PW8_AFtc_edit_dir.append([Dec,Inc,1.])
PW8_AFtc_edit_mean=pmag.fisher_mean(PW8_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW8_AFtc_dir,color='orange')
IPmag.iplotDI(PW8_AFtc_edit_dir,color='green')
IPmag.iplotDImean(PW8_AFtc_edit_mean['dec'],PW8_AFtc_edit_mean['inc'],
                  PW8_AFtc_edit_mean["alpha95"],color='green',marker='s',
                  label='PW8_edit')
```



Two AF samples from PW8 are added to the Mabogoapitse Hill Sill mean direction.

Mabogoapitse Hill Sill Combined Mean (PW7_ALL, PW8_AF)

```
In [29]: Mabogoapitse_Sill=[]
         Mabogoapitse_Sill = PW7_AFtc_edit_dir + PW7_tc_edit_dir + PW8_AFtc_edit_dir
         Mabogoapitse_Sill_mean=pmag.fisher_mean(Mabogoapitse_Sill)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(Mabogoapitse_Sill,color='blue')
         IPmag.iplotDImean(Mabogoapitse_Sill_mean['dec'],Mabogoapitse_Sill_mean['inc'],
                           Mabogoapitse_Sill_mean["alpha95"],color='b',marker='s',
                           label='Mabogoapitse_Sill_mean')
```




```
In [30]: Intrusion_mean_directions.loc['Mabogoapitse_Hill_Sill'] = pd.Series({
             'Intrusion_name':'Mabogoapitse_Hill_Sill',
             'sites_used':'PW7_ALL and PW8_AF',
             'site_lat':Site_Locations['LAT(WGS84)'][7],
             'site_long':Site_Locations['LONG(WGS84)'][7],
             'n':int(Mabogoapitse_Sill_mean['n']),
             'dec_tc':round(Mabogoapitse_Sill_mean['dec'],1),
             'inc_tc':round(Mabogoapitse_Sill_mean['inc'],1),
             'a_95':round(Mabogoapitse_Sill_mean['alpha95'],1),
             'k':round(Mabogoapitse_Sill_mean['k'],1),
             'dip_direction':0,
             'dip':0})
     Intrusion_mean_directions.ix['Mabogoapitse_Hill_Sill']

Out[30]: Intrusion_name     Mabogoapitse_Hill_Sill
     sites_used             PW7_ALL and PW8_AF
```

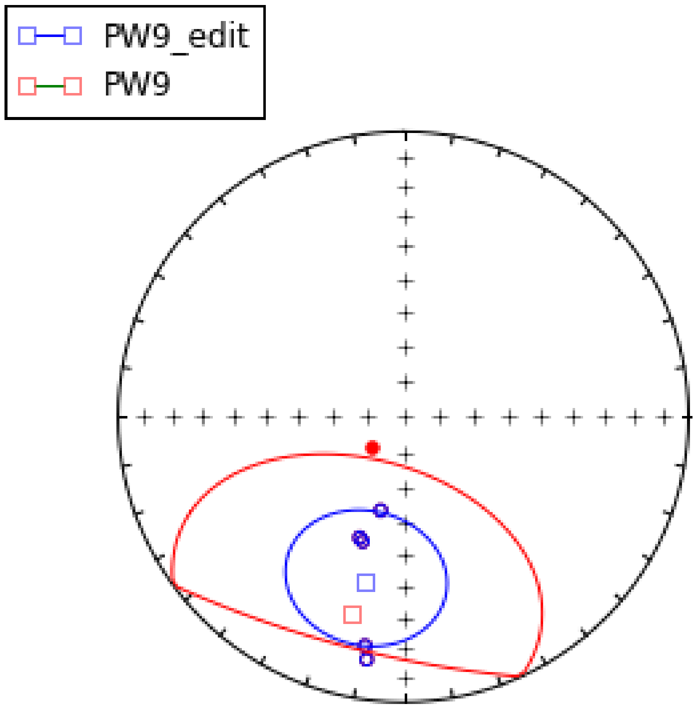
```
site_lat                          -24.47402
site_long                          25.59705
n                                         9
dec_tc                                184.6
inc_tc                                  5.1
a_95                                    9.1
k                                      33.2
date                                    NaN
date_error                              NaN
dip_direction                             0
dip                                       0
Name: Mabogoapitse_Hill_Sill, dtype: object
```

3.4.7 Semarule Hill Sill - PW9

Thermal demag

```
In [31]: PW9_tc_dir=[]
         for n in range(len(PW9_tc)):
             Dec,Inc=PW9_tc['specimen_dec'][n],PW9_tc['specimen_inc'][n]
             PW9_tc_dir.append([Dec,Inc,1.])
         PW9_tc_mean=pmag.fisher_mean(PW9_tc_dir)

         #Excluding one outlier point:
         PW9_tc_edit = PW9_tc
         PW9_tc_edit = PW9_tc_edit.drop(5)
         PW9_tc_edit.reset_index(inplace=True)

         PW9_tc_edit_dir=[]
         for n in range(len(PW9_tc_edit)):
             Dec,Inc=PW9_tc_edit['specimen_dec'][n],PW9_tc_edit['specimen_inc'][n]
             PW9_tc_edit_dir.append([Dec,Inc,1.])
         PW9_tc_edit_mean=pmag.fisher_mean(PW9_tc_edit_dir)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(PW9_tc_dir,color='r')
         IPmag.iplotDI(PW9_tc_edit_dir,color='b')
         IPmag.iplotDImean(PW9_tc_edit_mean['dec'],PW9_tc_edit_mean['inc'],
                           PW9_tc_edit_mean["alpha95"],color='b',marker='s',
                           label='PW9_edit')
         IPmag.iplotDImean(PW9_tc_mean['dec'],PW9_tc_mean['inc'],
                           PW9_tc_mean["alpha95"],color='r',marker='s',
```

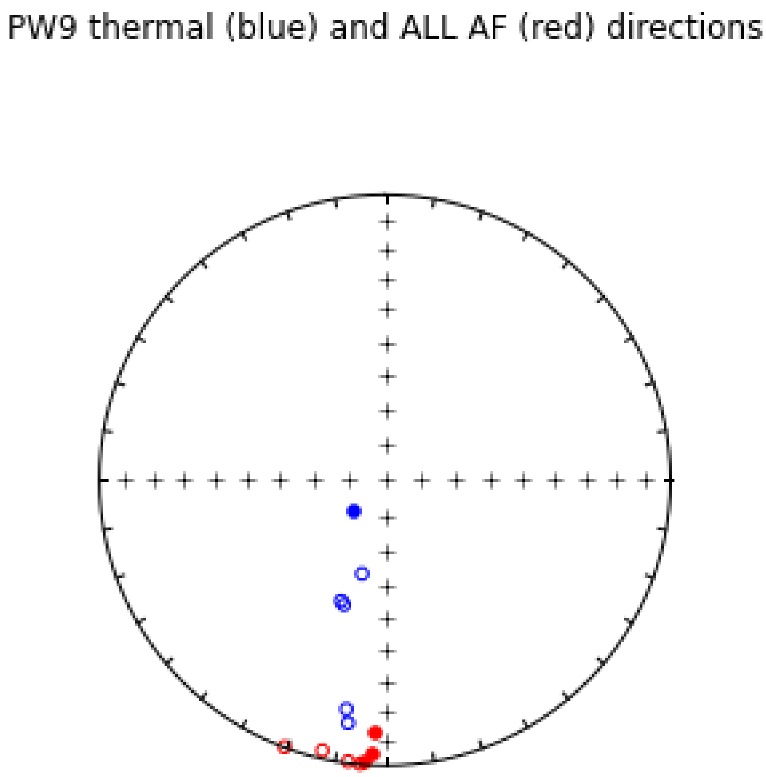
```
label='PW9')
```



AF demag

```
In [32]: PW9_AFdata=pd.read_csv('../Data/Botswana_AF/PW9/pmag_specimens.txt',
                                 sep='\t',header=1)
         PW9_AFtc = PW9_AFdata[PW9_AFdata['specimen_tilt_correction'] == 100]
         PW9_AFtc.reset_index(drop=True, inplace=True)

         PW9_AFtc_dir=[]
         for n in range(len(PW9_AFtc)):
             Dec,Inc=PW9_AFtc['specimen_dec'][n],PW9_AFtc['specimen_inc'][n],
             PW9_AFtc_dir.append([Dec,Inc,1.])
         PW9_AFtc_mean=pmag.fisher_mean(PW9_AFtc_dir)
```

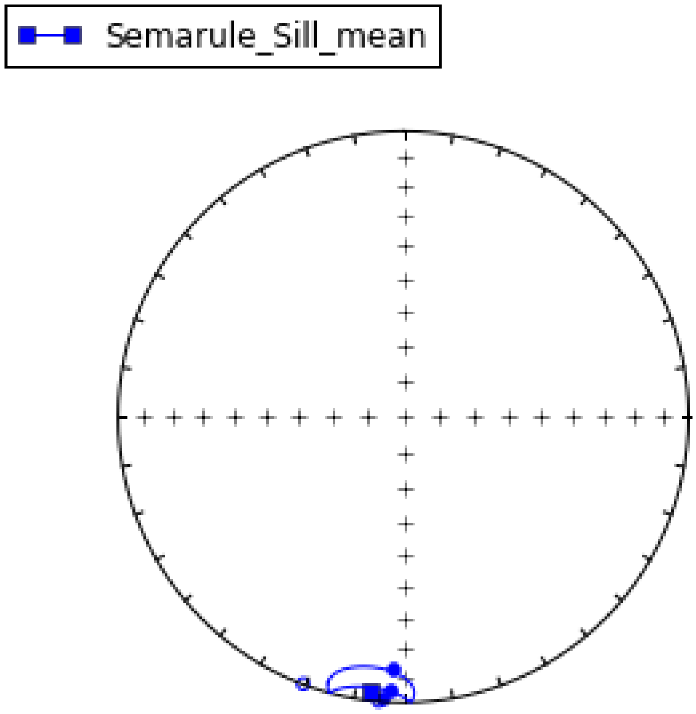
```
#Drop redundant specimens from same sample out of AF group
PW9_AFtc_edit = PW9_AFtc
PW9_AFtc_edit = PW9_AFtc_edit.drop(5)
PW9_AFtc_edit = PW9_AFtc_edit.drop(6)
PW9_AFtc_edit.reset_index(drop=True, inplace=True)
PW9_AFtc_edit_dir=[]
for n in range(len(PW9_AFtc_edit)):
    Dec,Inc=PW9_AFtc_edit['specimen_dec'][n],PW9_AFtc_edit['specimen_inc'][n],
    PW9_AFtc_edit_dir.append([Dec,Inc,1.])
PW9_AFtc_edit_mean=pmag.fisher_mean(PW9_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW9_tc_dir,color='blue')
IPmag.iplotDI(PW9_AFtc_dir,color='red')
plt.title('PW9 thermal (blue) and ALL AF (red) directions')
plt.show()
```



Combined Mean (PW9_AF) AF results are far more consistent than the thermal analyses. Two thermal samples, PW9-7 and PW9-8, agree roughly with their AF conterparts. However, overall the AF demagnetization appears to have been more effective in cleaning the samples and these data are favored for the mean that is added to the compilation.

```
In [33]: Semarule_Sill=[]
         Semarule_Sill = PW9_AFtc_edit_dir
         Semarule_Sill_mean=pmag.fisher_mean(Semarule_Sill)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(Semarule_Sill,color='blue')
         IPmag.iplotDImean(Semarule_Sill_mean['dec'],Semarule_Sill_mean['inc'],
```

```
Semarule_Sill_mean["alpha95"],color='b',marker='s',
label='Semarule_Sill_mean')
```




```
In [34]: Intrusion_mean_directions.loc['Semarule_Hill_Sill']=pd.Series({
             'Intrusion_name':'Semarule_Hill_Sill','sites_used':'PW9_ALL',
             'site_lat':Site_Locations['LAT(WGS84)'][8],
             'site_long':Site_Locations['LONG(WGS84)'][8],
             'n':int(Semarule_Sill_mean['n']),
             'dec_tc':round(Semarule_Sill_mean['dec'],1),
             'inc_tc':round(Semarule_Sill_mean['inc'],1),
             'a_95':round(Semarule_Sill_mean['alpha95'],1),
             'k':round(Semarule_Sill_mean['k'],1),
             'dip_direction':0,
             'dip':0})
     Intrusion_mean_directions.ix['Semarule_Hill_Sill']
```

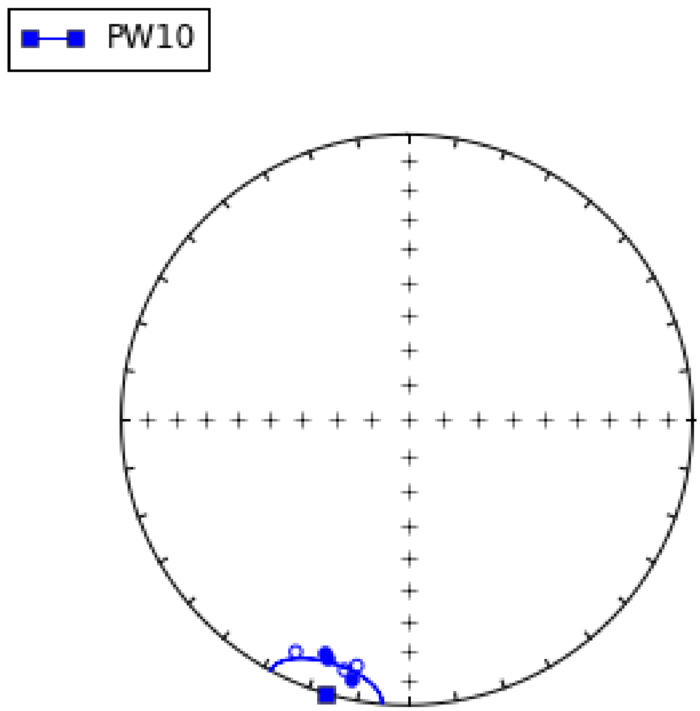
```
Out[34]: Intrusion_name    Semarule_Hill_Sill
         sites_used                   PW9_ALL
         site_lat                   -24.45268
         site_long                   25.57415
         n                                  5
         dec_tc                         186.8
         inc_tc                           3.8
         a_95                             9.1
         k                               72.2
         date                             NaN
         date_error                       NaN
         dip_direction                      0
         dip                                0
         Name: Semarule_Hill_Sill, dtype: object
```

3.4.8 Rapitsane Sill - PW10

Very good agreement and consistency in all samples.

Thermal demag

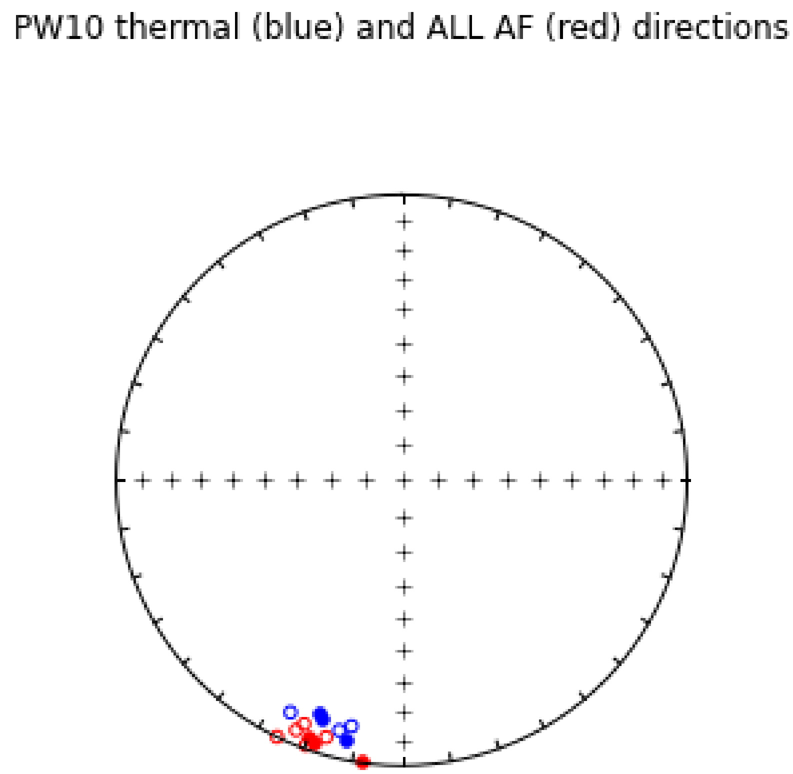
```
In [35]: PW10_tc_dir=[]
         for n in range(len(PW9_tc)):
             Dec,Inc=PW10_tc['specimen_dec'][n],PW10_tc['specimen_inc'][n]
             PW10_tc_dir.append([Dec,Inc,1.])
         PW10_tc_mean=pmag.fisher_mean(PW10_tc_dir)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(PW10_tc_dir,color='b')
         IPmag.iplotDImean(PW10_tc_mean['dec'],PW10_tc_mean['inc'],
                           PW10_tc_mean["alpha95"],color='b',marker='s',
                           label='PW10')
```



AF demag

```
In [36]: PW10_AFdata=pd.read_csv('../Data/Botswana_AF/PW10/pmag_specimens.txt',
                                sep='\t',header=1)
         PW10_AFtc = PW10_AFdata[PW10_AFdata['specimen_tilt_correction'] == 100]
         PW10_AFtc.reset_index(drop=True, inplace=True)

         PW10_AFtc_dir=[]
         for n in range(len(PW10_AFtc)):
             Dec,Inc=PW10_AFtc['specimen_dec'][n],PW10_AFtc['specimen_inc'][n]
             PW10_AFtc_dir.append([Dec,Inc,1.])
         PW10_AFtc_mean=pmag.fisher_mean(PW10_AFtc_dir)

         #Drop redundant specimens from same sample out of AF group
         PW10_AFtc_edit = PW10_AFtc
```

```
PW10_AFtc_edit = PW10_AFtc_edit.drop(0)
PW10_AFtc_edit = PW10_AFtc_edit.drop(1)
PW10_AFtc_edit = PW10_AFtc_edit.drop(2)
PW10_AFtc_edit = PW10_AFtc_edit.drop(3)
PW10_AFtc_edit = PW10_AFtc_edit.drop(4)
PW10_AFtc_edit = PW10_AFtc_edit.drop(5)
PW10_AFtc_edit.reset_index(drop=True, inplace=True)
PW10_AFtc_edit_dir=[]
for n in range(len(PW10_AFtc_edit)):
    Dec,Inc=PW10_AFtc_edit['specimen_dec'][n],PW10_AFtc_edit['specimen_inc'][n]
    PW10_AFtc_edit_dir.append([Dec,Inc,1.])
PW10_AFtc_edit_mean=pmag.fisher_mean(PW10_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW10_tc_dir,color='blue')
IPmag.iplotDI(PW10_AFtc_dir,color='red')
plt.title('PW10 thermal (blue) and ALL AF (red) directions')
plt.show()
```

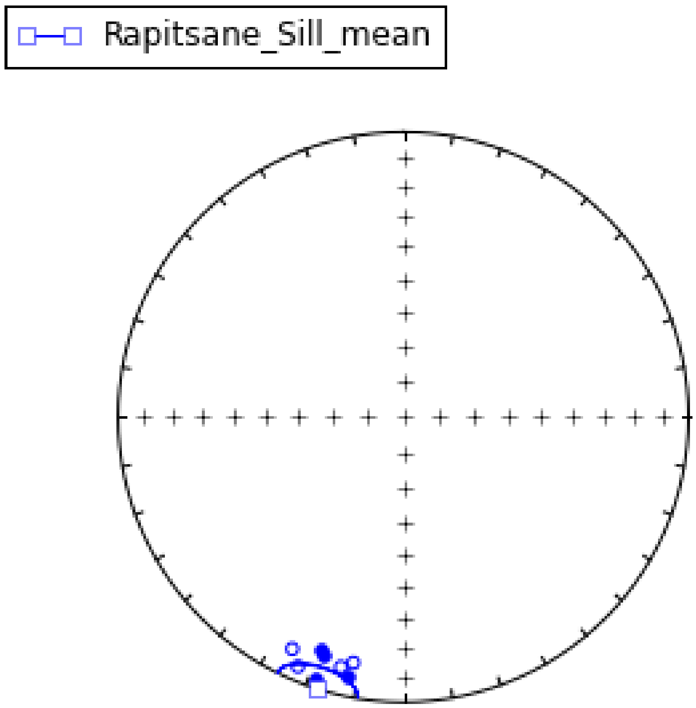


There is good agreement between all of the thermal and AF results. Two AF samples were not thermally demagnetized therefore their results are added to the mean.

Rapitsane Sill Mean (PW10_ALL)

```
In [37]: Rapitsane_Sill=[]
         Rapitsane_Sill = PW10_tc_dir + PW10_AFtc_edit_dir
         Rapitsane_Sill_mean=pmag.fisher_mean(Rapitsane_Sill)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(Rapitsane_Sill,color='blue')
         IPmag.iplotDImean(Rapitsane_Sill_mean['dec'],Rapitsane_Sill_mean['inc'],
                       Rapitsane_Sill_mean["alpha95"],color='b',marker='s',
```

```
                     label='Rapitsane_Sill_mean')
```



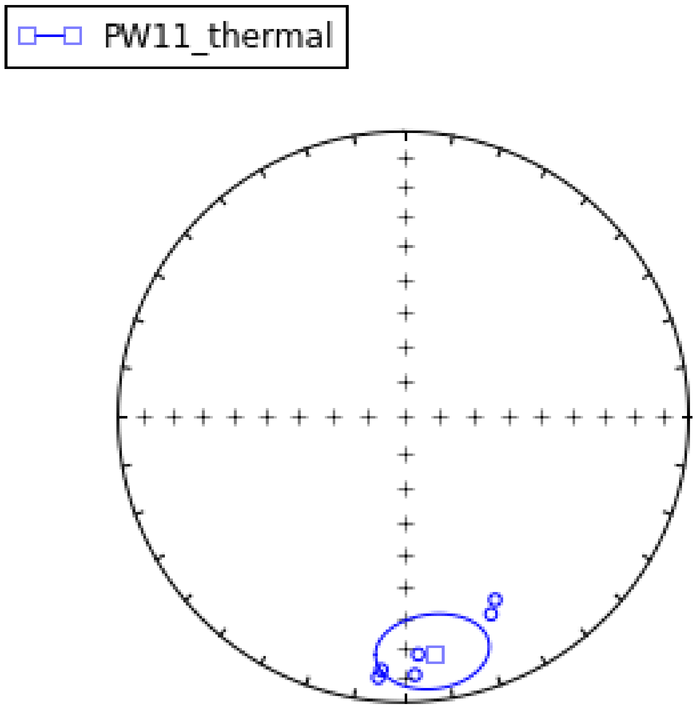

```
In [38]: Intrusion_mean_directions.loc['Rapitsane_Sill'] = pd.Series({'Intrusion_name':
                                 'Rapitsane_Sill','sites_used':'PW10_ALL',
                                 'site_lat':Site_Locations['LAT(WGS84)'][9],
                                 'site_long':Site_Locations['LONG(WGS84)'][9],
                                 'n':int(Rapitsane_Sill_mean['n']),
                                 'dec_tc':round(Rapitsane_Sill_mean['dec'],1),
                                 'inc_tc':round(Rapitsane_Sill_mean['inc'],1),
                                 'a_95':round(Rapitsane_Sill_mean['alpha95'],1),
                                 'k':round(Rapitsane_Sill_mean['k'],1),
                                 'dip_direction':330.0,
                                 'dip':7})
    Intrusion_mean_directions.ix['Rapitsane_Sill']
```

```
Out[38]: Intrusion_name    Rapitsane_Sill
     sites_used            PW10_ALL
     site_lat             -24.41968
     site_long             25.58463
     n                            8
     dec_tc                   197.7
     inc_tc                    -0.2
     a_95                       8.5
     k                         43.3
     date                       NaN
     date_error                 NaN
     dip_direction              330
     dip                          7
     Name: Rapitsane_Sill, dtype: object
```

3.4.9 Suping Sill - PW11,PW12,JP15,JP16

Thermal demag (PW11, not PW12) These results given an overall consistent direction.

```
In [39]: PW11_tc_dir=[]
     for n in range(len(PW11_tc)):
         Dec,Inc=PW11_tc['specimen_dec'][n],PW11_tc['specimen_inc'][n]
         PW11_tc_dir.append([Dec,Inc,1.])
     PW11_tc_mean=pmag.fisher_mean(PW11_tc_dir)

     fignum = 1
     plt.figure(num=fignum,figsize=(5,5))
     IPmag.iplotNET(1)
     IPmag.iplotDI(PW11_tc_dir,color='b')
     IPmag.iplotDImean(PW11_tc_mean['dec'],PW11_tc_mean['inc'],
                       PW11_tc_mean["alpha95"],color='b',marker='s',
                       label='PW11_thermal')
```

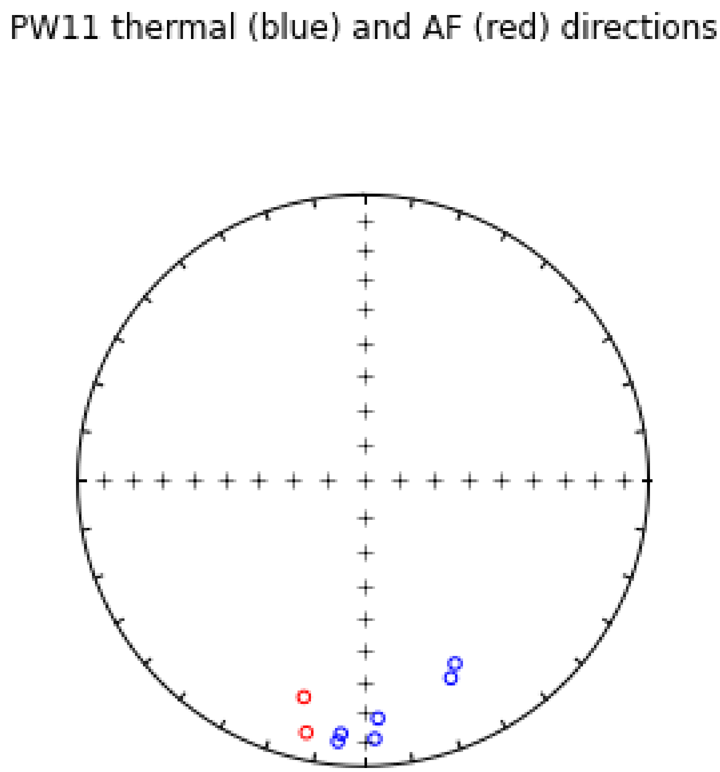


AF demag (PW11 and PW12)

PW11-AF

```
In [40]: PW11_AFdata=pd.read_csv('../Data/Botswana_AF/PW11/pmag_specimens.txt',
                                 sep='\t',header=1)
     PW11_AFtc = PW11_AFdata[PW11_AFdata['specimen_tilt_correction'] == 100]
     PW11_AFtc.reset_index(drop=True, inplace=True)

     PW11_AFtc_dir=[]
     for n in range(len(PW11_AFtc)):
         Dec,Inc=PW11_AFtc['specimen_dec'][n],PW11_AFtc['specimen_inc'][n]
         PW11_AFtc_dir.append([Dec,Inc,1.])
     PW11_AFtc_mean=pmag.fisher_mean(PW11_AFtc_dir)
```

```
#Drop redundant specimens from same samples in AF group (only 7&8 were new)
PW11_AFtc_edit = PW11_AFtc
PW11_AFtc_edit = PW11_AFtc_edit.drop(0)
PW11_AFtc_edit = PW11_AFtc_edit.drop(1)
PW11_AFtc_edit = PW11_AFtc_edit.drop(2)
PW11_AFtc_edit = PW11_AFtc_edit.drop(3)
PW11_AFtc_edit = PW11_AFtc_edit.drop(4)
PW11_AFtc_edit = PW11_AFtc_edit.drop(5)
PW11_AFtc_edit.reset_index(drop=True, inplace=True)
PW11_AFtc_edit_dir=[]
for n in range(len(PW11_AFtc_edit)):
    Dec,Inc=PW11_AFtc_edit['specimen_dec'][n],PW11_AFtc_edit['specimen_inc'][n]
    PW11_AFtc_edit_dir.append([Dec,Inc,1.])
PW11_AFtc_edit_mean=pmag.fisher_mean(PW11_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW11_tc_dir,color='blue')
IPmag.iplotDI(PW11_AFtc_edit_dir,color='red')
plt.title('PW11 thermal (blue) and AF (red) directions')
plt.show()
```

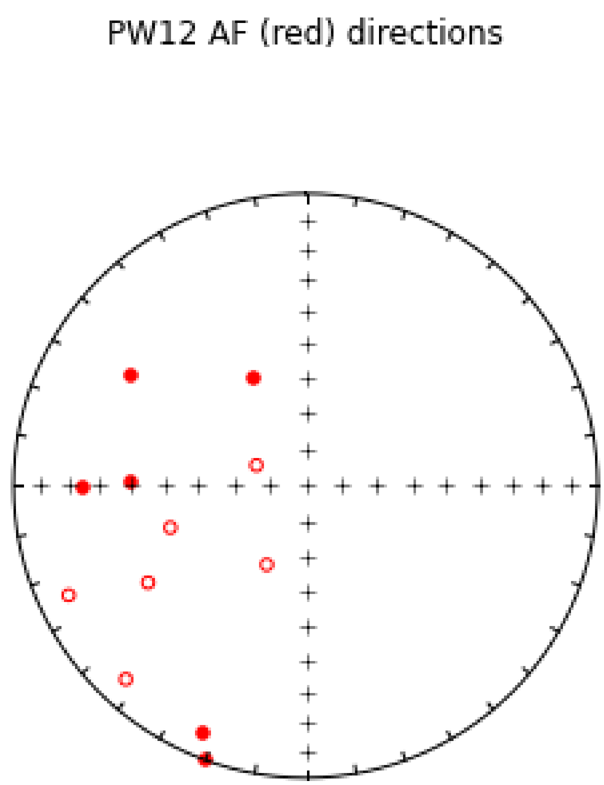


Red circles are the two additional AF samples from PW11.

PW12-AF

```
In [41]: PW12_AFdata=pd.read_csv('../Data/Botswana_AF/PW12/pmag_specimens.txt',
                                   sep='\t',header=1)
         PW12_AFtc = PW12_AFdata[PW12_AFdata['specimen_tilt_correction'] == 100]
         PW12_AFtc.reset_index(drop=True, inplace=True)

         PW12_AFtc_dir=[]
         for n in range(len(PW12_AFtc)):
             Dec,Inc=PW12_AFtc['specimen_dec'][n],PW12_AFtc['specimen_inc'][n]
             PW12_AFtc_dir.append([Dec,Inc,1.])
         PW12_AFtc_mean=pmag.fisher_mean(PW12_AFtc_dir)
```

```
PW12_AFtc_edit = PW12_AFtc
PW12_AFtc_edit = PW12_AFtc_edit.ix[3]
PW12_AFtc_edit_dir=[]
Dec,Inc=PW12_AFtc_edit['specimen_dec'],PW12_AFtc_edit['specimen_inc']
PW12_AFtc_edit_dir.append([Dec,Inc,1.])
PW12_AFtc_edit_mean=pmag.fisher_mean(PW12_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW12_AFtc_dir,color='red')
plt.title('PW12 AF (red) directions')
plt.show()
```



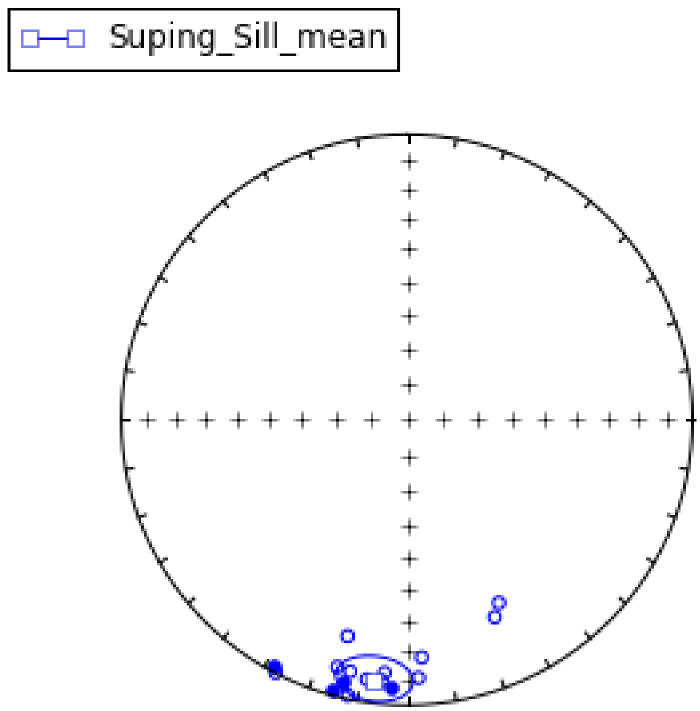
The PW12 AF data are scattered and are not used for the combined mean.

We also need to import the Pancake (2001) data for JP(15,16) so that it can be combined with the data from PW11/12; These sites sampled the same intrusion. We only import data from JP15 because JP16 yielded fairly scattered results from an outcrop that is slightly difficult to correlate.

```
In [42]: JP15_dir = pickle.load(open('../Data/Pickle/JP15','rb'))
```

Combined Mean (PW11_ALL, PW12_AF, JP(15,16,19))

```
In [43]: Suping_Sill=[]
         Suping_Sill = PW11_AFtc_edit_dir + PW11_tc_dir + JP15_dir
         Suping_Sill_mean=pmag.fisher_mean(Suping_Sill)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(Suping_Sill,color='blue')
         IPmag.iplotDImean(Suping_Sill_mean['dec'],Suping_Sill_mean['inc'],
                           Suping_Sill_mean["alpha95"],color='b',marker='s',
                           label='Suping_Sill_mean')
```



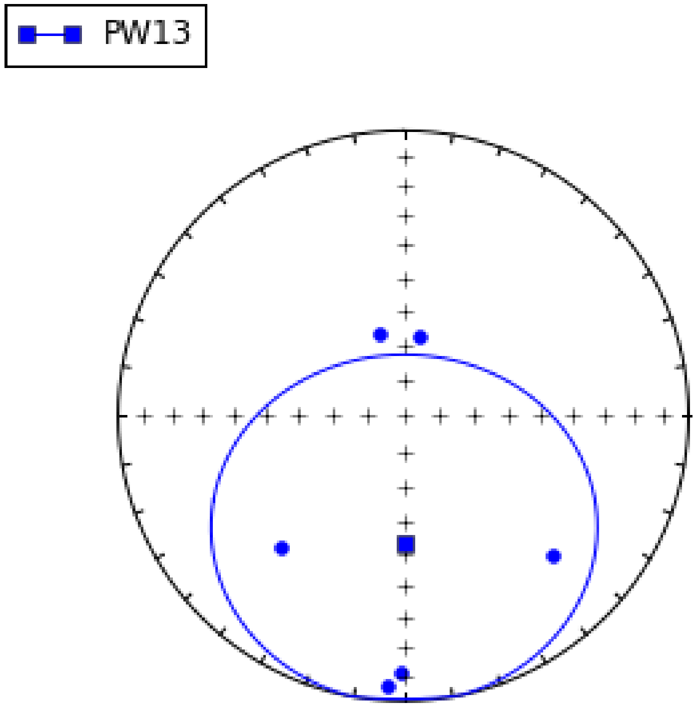

```
In [44]: Intrusion_mean_directions.loc['Suping_Sill'] = pd.Series({'Intrusion_name':
                                  'Suping_Sill',
                                  'sites_used':'PW11_ALL and JP15',
                                  'site_lat':Site_Locations['LAT(WGS84)'][10],
                                  'site_long':Site_Locations['LONG(WGS84)'][10],
                                  'n':int(Suping_Sill_mean['n']),
                                  'dec_tc':round(Suping_Sill_mean['dec'],1),
                                  'inc_tc':round(Suping_Sill_mean['inc'],1),
                                  'a_95':round(Suping_Sill_mean['alpha95'],1),
                                  'k':round(Suping_Sill_mean['k'],1),
                                  'dip_direction':281.2,
                                  'dip':10})
         Intrusion_mean_directions.ix['Suping_Sill']

Out[44]: Intrusion_name          Suping_Sill
         sites_used        PW11_ALL and JP15
```

```
site_lat                   -24.32765
site_long                   25.53224
n                                 16
dec_tc                         187.2
inc_tc                          -9.2
a_95                             8.4
k                               20.2
date                             NaN
date_error                       NaN
dip_direction                  281.2
dip                               10
Name: Suping_Sill, dtype: object
```

3.4.10 Gajong Donut Sill - PW13

Thermal demag

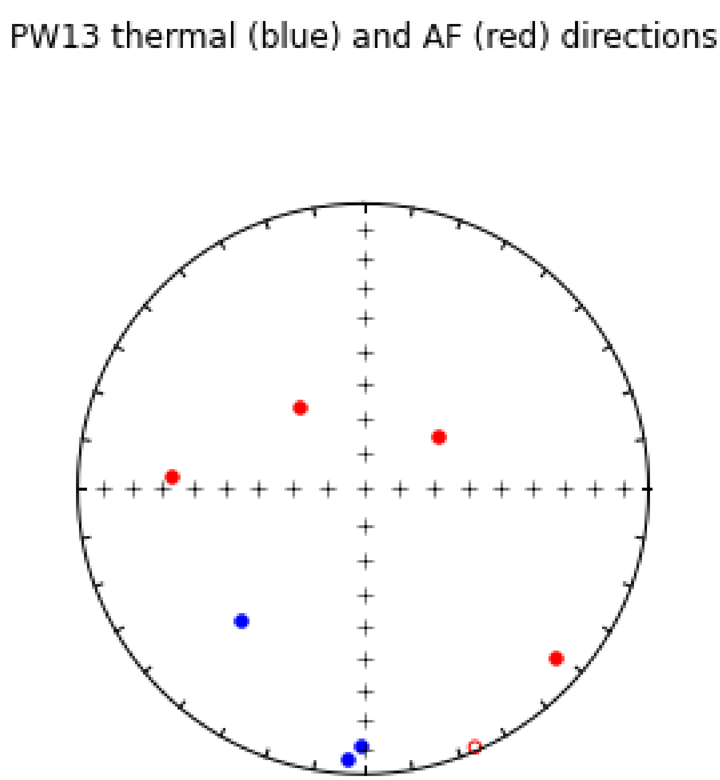
```
In [45]: PW13_tc_dir=[]
     for n in range(len(PW13_tc)):
         Dec,Inc=PW13_tc['specimen_dec'][n],PW13_tc['specimen_inc'][n]
         PW13_tc_dir.append([Dec,Inc,1.])
     PW13_tc_mean=pmag.fisher_mean(PW13_tc_dir)

     PW13_tc_edit = PW13_tc
     PW13_tc_edit = PW13_tc_edit.drop(0)
     PW13_tc_edit = PW13_tc_edit.drop(1)
     PW13_tc_edit = PW13_tc_edit.drop(2)
     PW13_tc_edit.reset_index(inplace=True)

     PW13_tc_edit_dir=[]
     for n in range(len(PW13_tc_edit)):
         Dec,Inc=PW13_tc_edit['specimen_dec'][n],PW13_tc_edit['specimen_inc'][n]
         PW13_tc_edit_dir.append([Dec,Inc,1.])
     PW13_tc_edit_mean=pmag.fisher_mean(PW13_tc_edit_dir)

     fignum = 1
     plt.figure(num=fignum,figsize=(5,5))
     IPmag.iplotNET(1)
     IPmag.iplotDI(PW13_tc_dir,color='b')
     IPmag.iplotDImean(PW13_tc_mean['dec'],PW13_tc_mean['inc'],
                       PW13_tc_mean["alpha95"],color='b',marker='s',label='PW13')
```



PW13 AF demag and combined thermal/AF samples

```
In [46]: PW13_AFdata=pd.read_csv('../Data/Botswana_AF/PW13/pmag_specimens.txt',
                                 sep='\t',header=1)
         PW13_AFtc = PW13_AFdata[PW13_AFdata['specimen_tilt_correction'] == 100]
         PW13_AFtc.reset_index(drop=True, inplace=True)

         PW13_AFtc_dir=[]
         for n in range(len(PW13_AFtc)):
             Dec,Inc=PW13_AFtc['specimen_dec'][n],PW13_AFtc['specimen_inc'][n]
             PW13_AFtc_dir.append([Dec,Inc,1.])
         PW13_AFtc_mean=pmag.fisher_mean(PW13_AFtc_dir)

         #Drop redundant specimens from same sample out of AF group
         PW13_AFtc_edit = PW13_AFtc
```

```
PW13_AFtc_edit = PW13_AFtc_edit.drop(4)
PW13_AFtc_edit = PW13_AFtc_edit.drop(5)
PW13_AFtc_edit = PW13_AFtc_edit.drop(6)
PW13_AFtc_edit.reset_index(drop=True, inplace=True)
PW13_AFtc_edit_dir=[]
for n in range(len(PW13_AFtc_edit)):
    Dec,Inc=PW13_AFtc_edit['specimen_dec'][n],PW13_AFtc_edit['specimen_inc'][n]
    PW13_AFtc_edit_dir.append([Dec,Inc,1.])
PW13_AFtc_edit_mean=pmag.fisher_mean(PW13_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW13_tc_edit_dir,color='blue')
IPmag.iplotDI(PW13_AFtc_edit_dir,color='red')
plt.title('PW13 thermal (blue) and AF (red) directions')
plt.show()
```

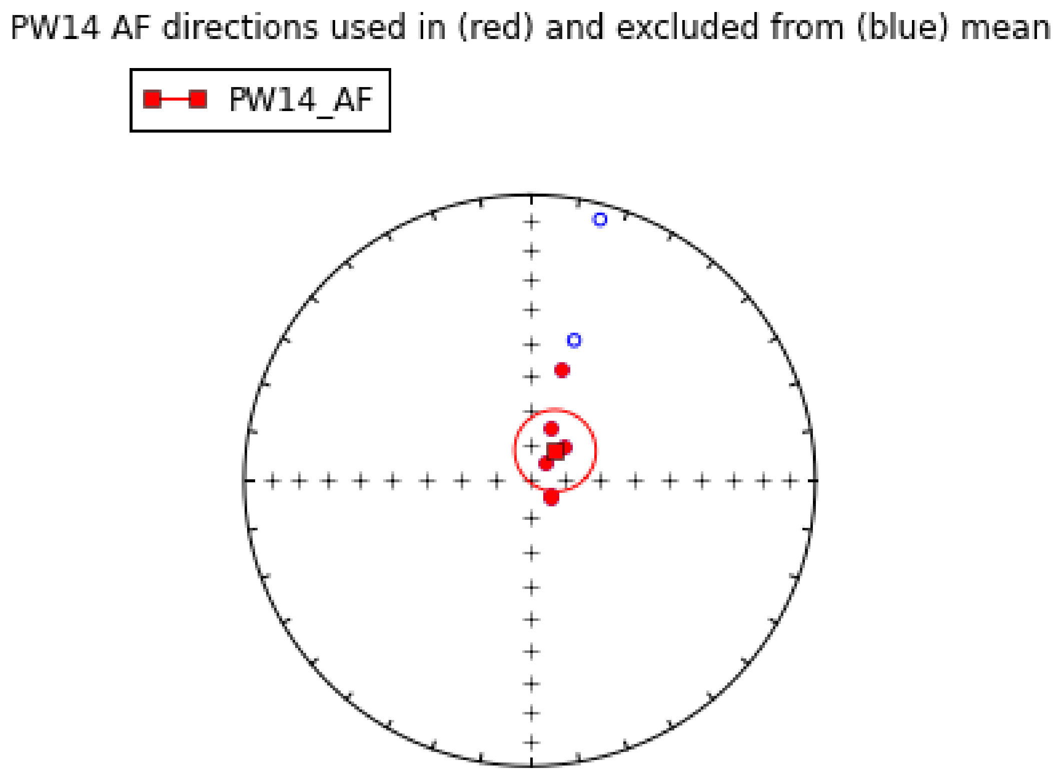


This site does not reveal a consistent grouping of directions and is not used in the compilation.

3.4.11 Mogatelwane 1 Sill - PW14

Samples from PW14 were only AF demagnetized. Results appear to be consistent, but with a direction far different from that which is typical of the Umkondo LIP.

```
In [47]: PW14_AFdata=pd.read_csv('../Data/Botswana_AF/PW14/pmag_specimens.txt',
                                  sep='\t',header=1)
         PW14_AFtc = PW14_AFdata[PW14_AFdata['specimen_tilt_correction'] == 100]
         PW14_AFtc.reset_index(drop=True, inplace=True)

         PW14_AFtc_dir=[]
         for n in range(len(PW14_AFtc)):
             Dec,Inc=PW14_AFtc['specimen_dec'][n],PW14_AFtc['specimen_inc'][n]
             PW14_AFtc_dir.append([Dec,Inc,1.])
         PW14_AFtc_mean=pmag.fisher_mean(PW14_AFtc_dir)

         #Drop redundant specimens from same sample out of AF group
         PW14_AFtc_edit = PW14_AFtc
         PW14_AFtc_edit = PW14_AFtc_edit.drop(0)
         PW14_AFtc_edit = PW14_AFtc_edit.drop(1)
         PW14_AFtc_edit.reset_index(drop=True, inplace=True)
         PW14_AFtc_edit_dir=[]
         for n in range(len(PW14_AFtc_edit)):
             Dec,Inc=PW14_AFtc_edit['specimen_dec'][n],PW14_AFtc_edit['specimen_inc'][n]
             PW14_AFtc_edit_dir.append([Dec,Inc,1.])
         PW14_AFtc_edit_mean=pmag.fisher_mean(PW14_AFtc_edit_dir)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(PW14_AFtc_dir,color='blue')
         IPmag.iplotDI(PW14_AFtc_edit_dir,color='red')
         IPmag.iplotDImean(PW14_AFtc_edit_mean['dec'],PW14_AFtc_edit_mean['inc'],
                           PW14_AFtc_edit_mean["alpha95"],color='r',marker='s',label='PW14_AF')
         plt.title('PW14 AF directions used in (red) and excluded from (blue) mean')
         plt.show()
```




```
In [48]: #Load unknown/younger/older intrusion table, so that PW18_thermal can be added
         unknown_intrusions = pickle.load(open('../Data/Pickle/unknown_intrusions','rb'))
```

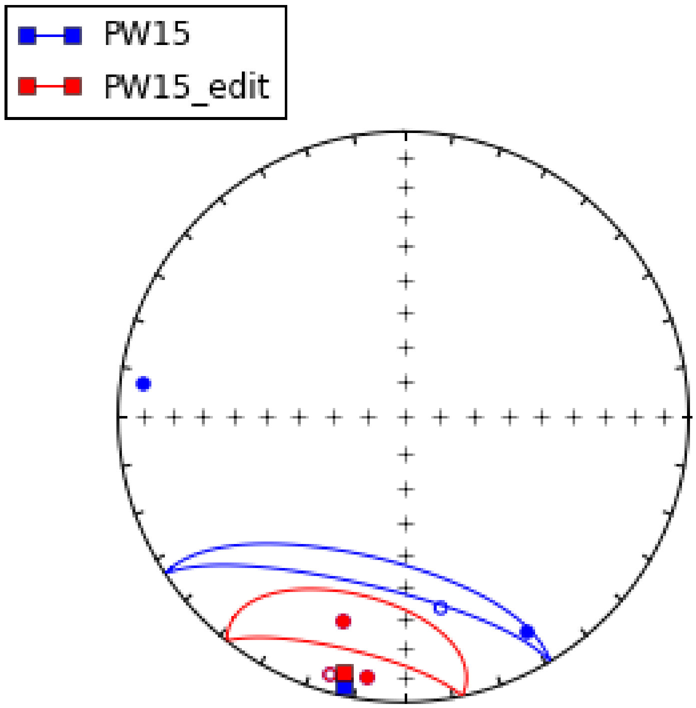
```
In [49]: #add to unknown intrusion table
         unknown_intrusions.loc['Mogatelwane_1_Sill'] = pd.Series({'site_ID':'PW14_AF',
                                         'site_lat':Site_Locations['LAT(WGS84)'][13],
                                         'site_long':Site_Locations['LONG(WGS84)'][13],
                                         'n':PW14_AFtc_edit_mean['n'],
                                         'dec_tc':round(PW14_AFtc_edit_mean['dec'],1),
                                         'inc_tc':round(PW14_AFtc_edit_mean['inc'],1),
                                         'a_95':round(PW14_AFtc_edit_mean['alpha95'],1),
                                         'k':round(PW14_AFtc_edit_mean['k'],1)})
```

3.4.12 Mogatelwane 2 Sill - PW15,PW16,PW17

PW16 and PW17 are from two different rotated blocks that were sampled for paleointensity but that cannot be used for paleodirectional analysis and are not included in this analysis.

Thermal demag (PW15 only)

```
In [50]: PW15_tc_dir=[]
         for n in range(len(PW15_tc)):
             Dec,Inc=PW15_tc['specimen_dec'][n],PW15_tc['specimen_inc'][n]
             PW15_tc_dir.append([Dec,Inc,1.])
         PW15_tc_mean=pmag.fisher_mean(PW15_tc_dir)

         PW15_tc_edit = PW15_tc
         PW15_tc_edit = PW15_tc_edit.drop(0)
         PW15_tc_edit = PW15_tc_edit.drop(2)
         PW15_tc_edit = PW15_tc_edit.drop(5)
         PW15_tc_edit.reset_index(inplace=True)

         PW15_tc_edit_dir=[]
         for n in range(len(PW15_tc_edit)):
             Dec,Inc=PW15_tc_edit['specimen_dec'][n],PW15_tc_edit['specimen_inc'][n]
             PW15_tc_edit_dir.append([Dec,Inc,1.])
         PW15_tc_edit_mean=pmag.fisher_mean(PW15_tc_edit_dir)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(PW15_tc_dir,color='b')
         IPmag.iplotDImean(PW15_tc_mean['dec'],PW15_tc_mean['inc'],
                           PW15_tc_mean["alpha95"],color='b',marker='s',
                           label='PW15')
         IPmag.iplotDI(PW15_tc_edit_dir,color='r')
         IPmag.iplotDImean(PW15_tc_edit_mean['dec'],PW15_tc_edit_mean['inc'],
                           PW15_tc_edit_mean["alpha95"],color='r',marker='s',
                           label='PW15_edit')
```

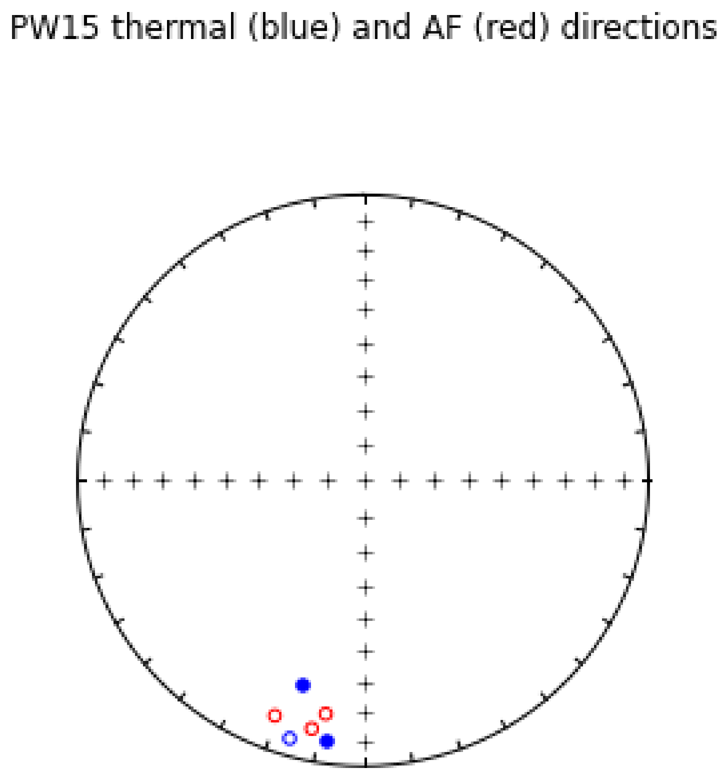


AF demag (PW15)

PW15_AF

```
In [51]: PW15_AFdata=pd.read_csv('../Data/Botswana_AF/PW15/pmag_specimens.txt',
                                 sep='\t',header=1)
         PW15_AFtc = PW15_AFdata[PW15_AFdata['specimen_tilt_correction'] == 100]
         PW15_AFtc.reset_index(drop=True, inplace=True)

         PW15_AFtc_dir=[]
         for n in range(len(PW15_AFtc)):
             Dec,Inc=PW15_AFtc['specimen_dec'][n],PW15_AFtc['specimen_inc'][n]
             PW15_AFtc_dir.append([Dec,Inc,1.])
         PW15_AFtc_mean=pmag.fisher_mean(PW15_AFtc_dir)
```

```
#Drop redundant specimens from same sample out of AF group
PW15_AFtc_edit = PW15_AFtc
PW15_AFtc_edit = PW15_AFtc_edit.drop(1)
PW15_AFtc_edit = PW15_AFtc_edit.drop(3)
PW15_AFtc_edit = PW15_AFtc_edit.drop(4)
PW15_AFtc_edit = PW15_AFtc_edit.drop(6)
PW15_AFtc_edit = PW15_AFtc_edit.drop(7)
PW15_AFtc_edit.reset_index(drop=True, inplace=True)
PW15_AFtc_edit_dir=[]
for n in range(len(PW15_AFtc_edit)):
    Dec,Inc=PW15_AFtc_edit['specimen_dec'][n],PW15_AFtc_edit['specimen_inc'][n]
    PW15_AFtc_edit_dir.append([Dec,Inc,1.])
PW15_AFtc_edit_mean=pmag.fisher_mean(PW15_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW15_tc_edit_dir,color='blue')
IPmag.iplotDI(PW15_AFtc_edit_dir,color='red')
plt.title('PW15 thermal (blue) and AF (red) directions')
plt.show()
```

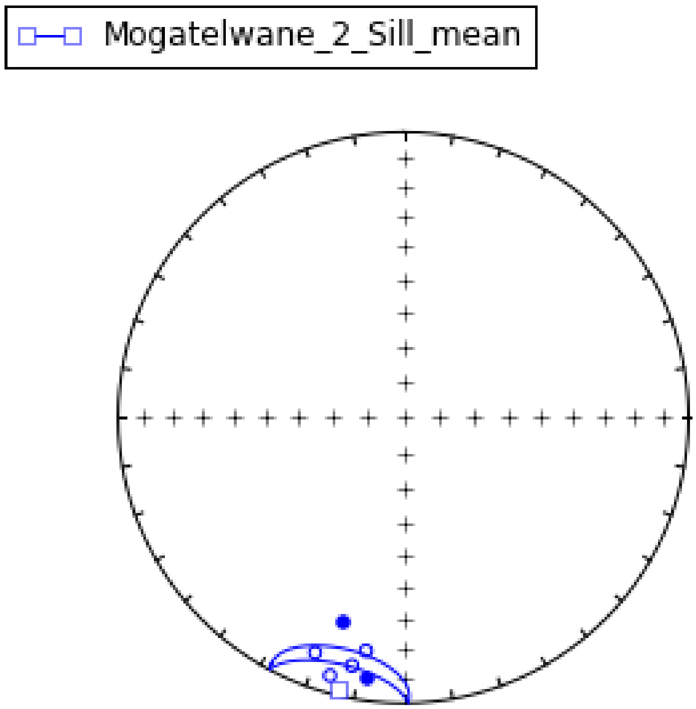


There is rough agreement between six different samples, 3 AF and 3 thermal results. Results from other samples are scattered.

Mogatelwane 2 Sill Combined mean (PW15_ALL)

```
In [52]: Mogatelwane_2_Sill=[]
         Mogatelwane_2_Sill = PW15_AFtc_edit_dir + PW15_tc_edit_dir
         Mogatelwane_2_Sill_mean=pmag.fisher_mean(Mogatelwane_2_Sill)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(Mogatelwane_2_Sill,color='blue')
         IPmag.iplotDImean(Mogatelwane_2_Sill_mean['dec'],
                           Mogatelwane_2_Sill_mean['inc'],
```

```
Mogatelwane_2_Sill_mean["alpha95"],color='b',
marker='s',label='Mogatelwane_2_Sill_mean')
```



The scatter of this mean direction is large but is close to the Umkondo direction. This will be included in the compilation but may be excised later in the analysis due to the large a95.

```
In [53]: Intrusion_mean_directions.loc['Mogatelwane 2 Sill'] = pd.Series({'Intrusion_name':
                                 'Mogatelwane 2 Sill',
                                 'sites_used':'PW15_ALL',
                                 'site_lat':Site_Locations['LAT(WGS84)'][14],
                                 'site_long':Site_Locations['LONG(WGS84)'][14],
                                 'n':int(Mogatelwane_2_Sill_mean['n']),
                                 'dec_tc':round(Mogatelwane_2_Sill_mean['dec'],1),
                                 'inc_tc':round(Mogatelwane_2_Sill_mean['inc'],1),
                                 'a_95':round(Mogatelwane_2_Sill_mean['alpha95'],1),
                                 'k':round(Mogatelwane_2_Sill_mean['k'],1),
```

```
                                'dip_direction':260.3,
                                'dip':4})
     Intrusion_mean_directions.ix['Mogatelwane 2 Sill']
```

```
Out[53]: Intrusion_name    Mogatelwane 2 Sill
     sites_used                  PW15_ALL
     site_lat                   -24.18042
     site_long                   25.69191
     n                                  6
     dec_tc                         193.5
     inc_tc                          -2.8
     a_95                            14.7
     k                               21.6
     date                             NaN
     date_error                       NaN
     dip_direction                  260.3
     dip                                4
     Name: Mogatelwane 2 Sill, dtype: object
```

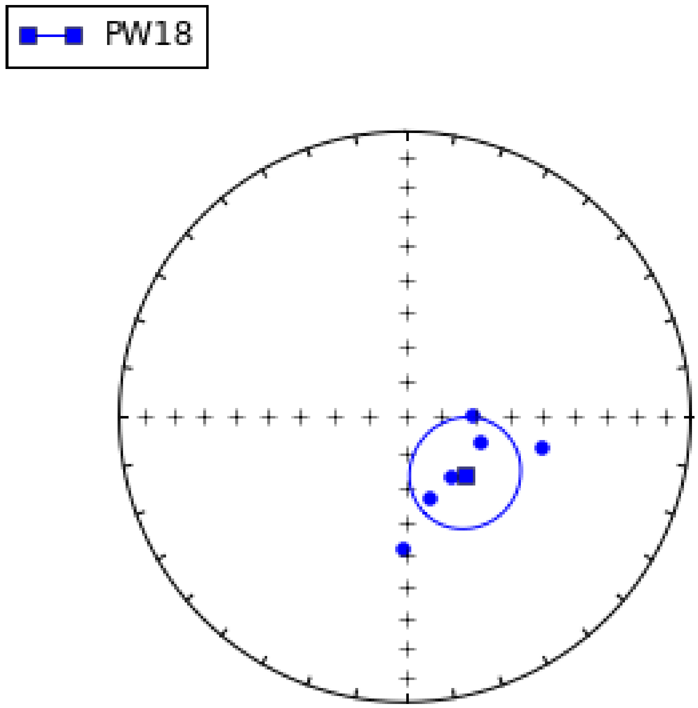
3.4.13 Molepolole Prison Quarry Sill - PW18,PW19

Some of these samples appear to carry a Karoo LIP direction that is opposite polarity to the Karoo dike reported in Gose et al. (2006) and significantly different from the Umkondo LIP magnetization direction.

Thermal demag

PW18_thermal

```
In [54]: PW18_tc_dir=[]
     for n in range(len(PW18_tc)):
         Dec,Inc=PW18_tc['specimen_dec'][n],PW18_tc['specimen_inc'][n]
         PW18_tc_dir.append([Dec,Inc,1.])
     PW18_tc_mean=pmag.fisher_mean(PW18_tc_dir)

     fignum = 1
     plt.figure(num=fignum,figsize=(5,5))
     IPmag.iplotNET(1)
     IPmag.iplotDI(PW18_tc_dir,color='b')
     IPmag.iplotDImean(PW18_tc_mean['dec'],PW18_tc_mean['inc'],
                   PW18_tc_mean["alpha95"],color='b',marker='s',
                   label='PW18')
```

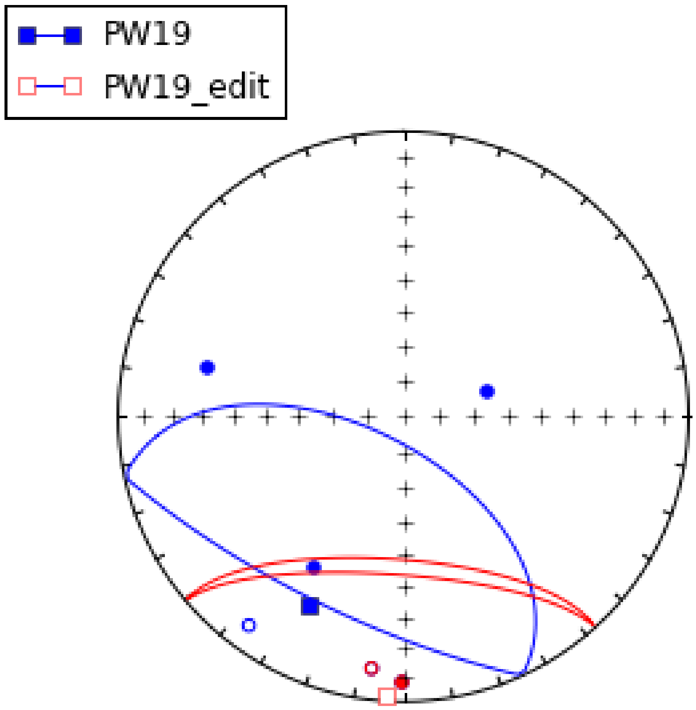


Probably a Karoo direction. Due to complications that can be seen below, this direction could be an overprint on an Umkondo-aged sill.

PW19_Thermal

```
In [55]: PW19_tc_dir=[]
         for n in range(len(PW19_tc)):
             Dec,Inc=PW19_tc['specimen_dec'][n],PW19_tc['specimen_inc'][n],
             PW19_tc_dir.append([Dec,Inc,1.])
         PW19_tc_mean=pmag.fisher_mean(PW19_tc_dir)

         PW19_tc_edit = PW19_tc
         PW19_tc_edit = PW19_tc_edit.drop(0)
         PW19_tc_edit = PW19_tc_edit.drop(2)
         PW19_tc_edit = PW19_tc_edit.drop(4)
         PW19_tc_edit = PW19_tc_edit.drop(5)
```

```
PW19_tc_edit.reset_index(inplace=True)

PW19_tc_edit_dir=[]
for n in range(len(PW19_tc_edit)):
    Dec,Inc=PW19_tc_edit['specimen_dec'][n],PW19_tc_edit['specimen_inc'][n]
    PW19_tc_edit_dir.append([Dec,Inc,1.])
PW19_tc_edit_mean=pmag.fisher_mean(PW19_tc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW19_tc_dir,color='b')
IPmag.iplotDImean(PW19_tc_mean['dec'],PW19_tc_mean['inc'],
                  PW19_tc_mean["alpha95"],color='b',marker='s',
                  label='PW19')
IPmag.iplotDI(PW19_tc_edit_dir,color='r')
IPmag.iplotDImean(PW19_tc_edit_mean['dec'],PW19_tc_edit_mean['inc'],
                  PW19_tc_edit_mean["alpha95"],color='r',marker='s',
                  label='PW19_edit')
```

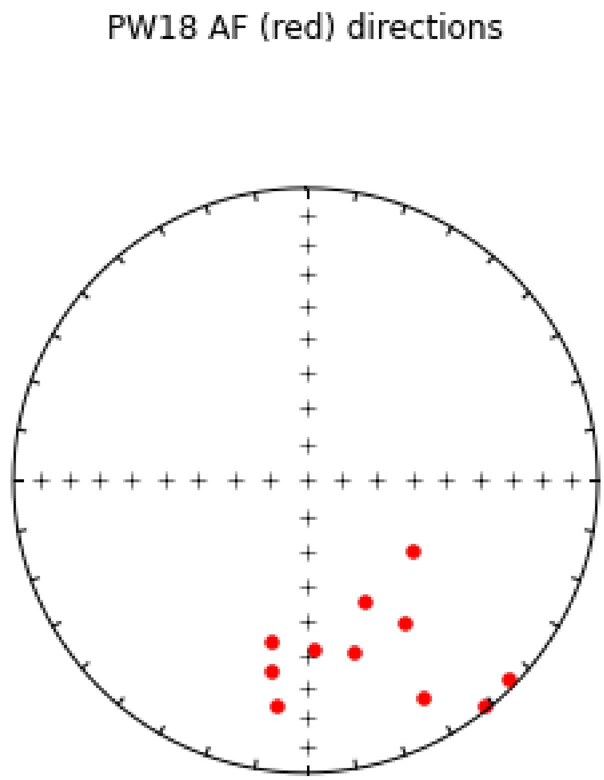


AF demag (PW18_AF and PW19_AF)

PW18_AF

```
In [56]: PW18_AFdata=pd.read_csv('../Data/Botswana_AF/PW18/pmag_specimens.txt',
                                 sep='\t',header=1)
     PW18_AFtc = PW18_AFdata[PW18_AFdata['specimen_tilt_correction'] == 100]
     PW18_AFtc.reset_index(drop=True, inplace=True)

     PW18_AFtc_dir=[]
     for n in range(len(PW18_AFtc)):
         Dec,Inc=PW18_AFtc['specimen_dec'][n],PW18_AFtc['specimen_inc'][n]
         PW18_AFtc_dir.append([Dec,Inc,1.])
     PW18_AFtc_mean=pmag.fisher_mean(PW18_AFtc_dir)
```

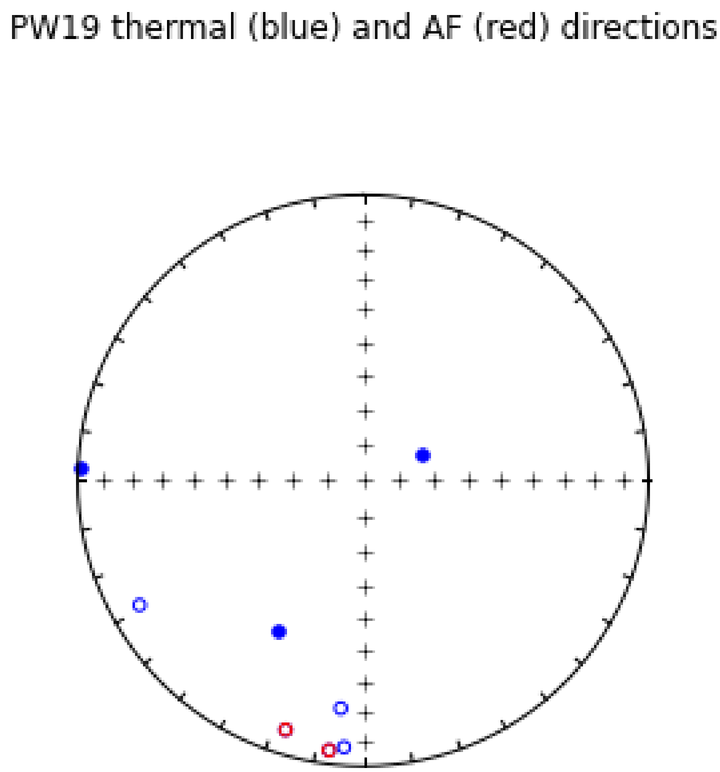
```
#Drop redundant specimens from same sample out of AF group.
PW18_AFtc_edit = PW18_AFtc
PW18_AFtc_edit = PW18_AFtc_edit.drop(4)
PW18_AFtc_edit = PW18_AFtc_edit.drop(5)
PW18_AFtc_edit = PW18_AFtc_edit.drop(12)
PW18_AFtc_edit.reset_index(drop=True, inplace=True)
PW18_AFtc_edit_dir=[]
for n in range(len(PW18_AFtc_edit)):
    Dec,Inc=PW18_AFtc_edit['specimen_dec'][n],PW18_AFtc_edit['specimen_inc'][n]
    PW18_AFtc_edit_dir.append([Dec,Inc,1.])
PW18_AFtc_edit_mean=pmag.fisher_mean(PW18_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW18_AFtc_edit_dir,color='red')
plt.title('PW18 AF (red) directions')
plt.show()
```



PW19_AF

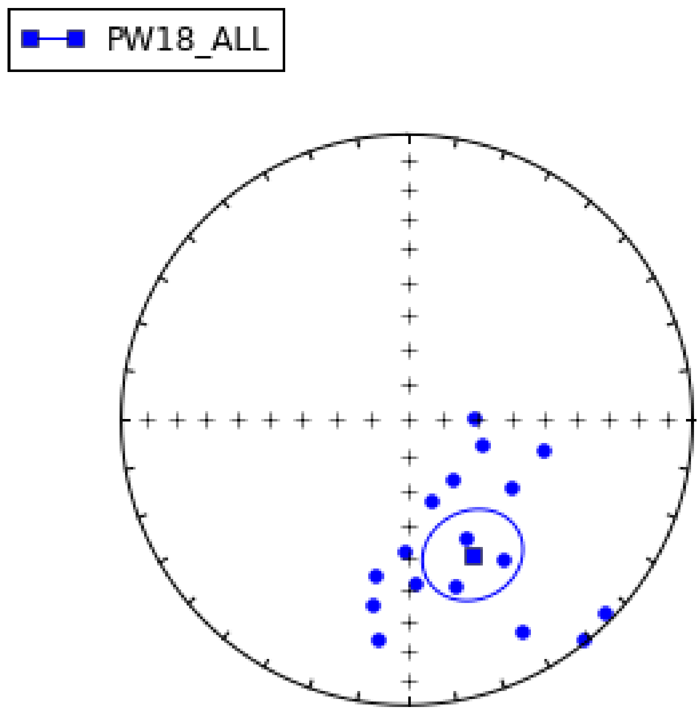
```
In [57]: PW19_AFdata=pd.read_csv('../Data/Botswana_AF/PW19/pmag_specimens.txt',
                                sep='\t',header=1)
         PW19_AFtc = PW19_AFdata[PW19_AFdata['specimen_tilt_correction'] == 100]
         PW19_AFtc.reset_index(drop=True, inplace=True)

         PW19_AFtc_dir=[]
         for n in range(len(PW19_AFtc)):
             Dec,Inc=PW19_AFtc['specimen_dec'][n],PW19_AFtc['specimen_inc'][n]
             PW19_AFtc_dir.append([Dec,Inc,1.])
         PW19_AFtc_mean=pmag.fisher_mean(PW19_AFtc_dir)

         #Drop redundant specimens from same sample out of AF group.
```

```
#Dropped even more samples because directions were scattered,...
#...only 2 AF directions being combined with thermal data.
PW19_AFtc_edit = PW19_AFtc
PW19_AFtc_edit = PW19_AFtc_edit.drop(3)
PW19_AFtc_edit = PW19_AFtc_edit.drop(5)
#PW19_AFtc_edit = PW19_AFtc_edit.drop(7)
PW19_AFtc_edit = PW19_AFtc_edit.drop(0)
PW19_AFtc_edit = PW19_AFtc_edit.drop(1)
PW19_AFtc_edit = PW19_AFtc_edit.drop(4)
PW19_AFtc_edit = PW19_AFtc_edit.drop(6)
PW19_AFtc_edit.reset_index(drop=True, inplace=True)
PW19_AFtc_edit_dir=[]
for n in range(len(PW19_AFtc_edit)):
    Dec,Inc=PW19_AFtc_edit['specimen_dec'][n],PW19_AFtc_edit['specimen_inc'][n]
    PW19_AFtc_edit_dir.append([Dec,Inc,1.])
PW19_AFtc_edit_mean=pmag.fisher_mean(PW19_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW19_AFtc_dir,color='blue')
IPmag.iplotDI(PW19_AFtc_edit_dir,color='red')
plt.title('PW19 thermal (blue) and AF (red) directions')
plt.show()
```

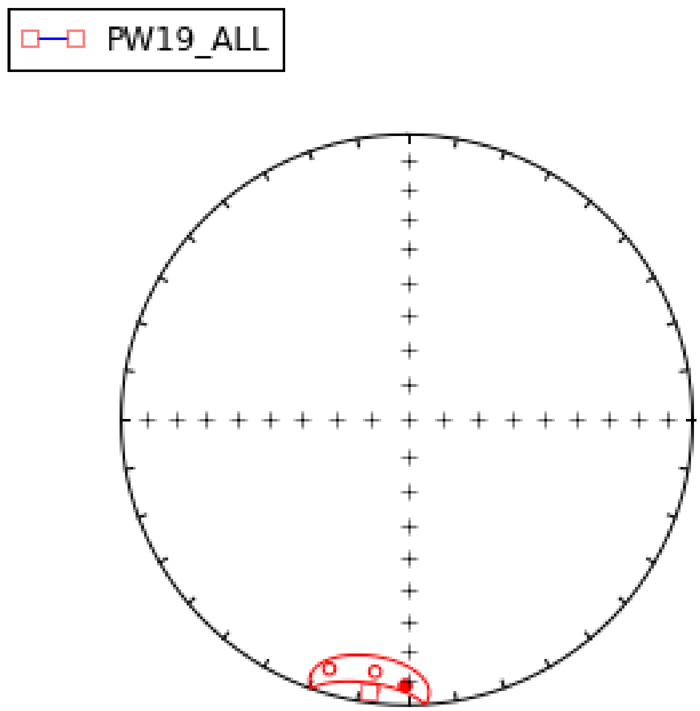


Combined Molepolole prison quarry sill

```
In [58]: PW18_ALL = PW18_tc_dir + PW18_AFtc_edit_dir
         PW18_ALL_mean=pmag.fisher_mean(PW18_ALL)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(PW18_ALL,color='blue')
         IPmag.iplotDImean(PW18_ALL_mean['dec'],PW18_ALL_mean['inc'],
                           PW18_ALL_mean["alpha95"],color='b',marker='s',
                           label='PW18_ALL')
```

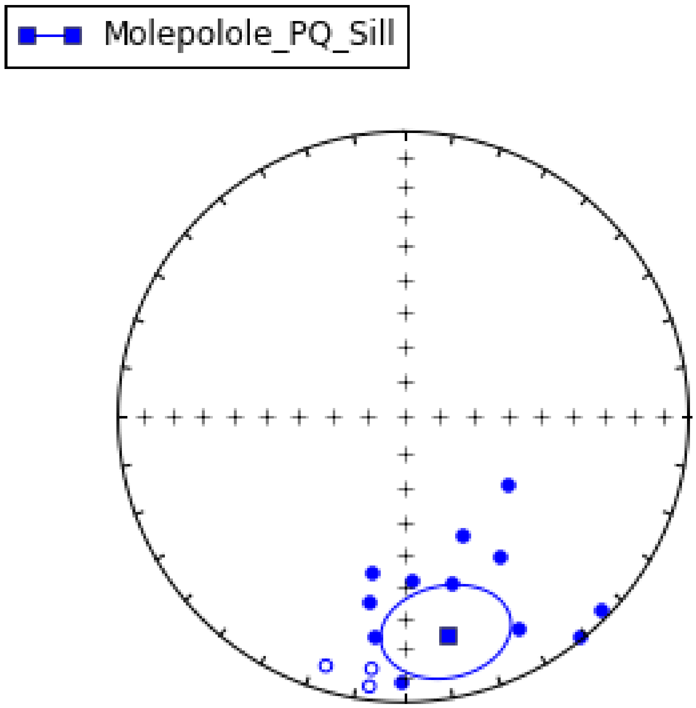



```
In [59]: PW19_ALL = PW19_tc_edit_dir + PW19_AFtc_edit_dir
         PW19_ALL_mean=pmag.fisher_mean(PW19_ALL)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(PW19_ALL,color='red')
         IPmag.iplotDImean(PW19_ALL_mean['dec'],PW19_ALL_mean['inc'],
                           PW19_ALL_mean["alpha95"],color='r',marker='s',
                           label='PW19_ALL')
```



If we consider the PW18_AF samples directions as Umkondo in origin, then we would dismiss PW18_thermal directions and add PW18_AF to PW19_ALL. This creates a wide spread of directions with a mediocre mean.

```
In [60]: Molepolole_PQ_Sill=[]
         Molepolole_PQ_Sill = PW19_ALL + PW18_AFtc_edit_dir
         Molepolole_PQ_Sill_mean=pmag.fisher_mean(Molepolole_PQ_Sill)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(Molepolole_PQ_Sill,color='blue')
         IPmag.iplotDImean(Molepolole_PQ_Sill_mean['dec'],
                           Molepolole_PQ_Sill_mean['inc'],
                           Molepolole_PQ_Sill_mean["alpha95"],color='b',marker='s',
                           label='Molepolole_PQ_Sill')
```



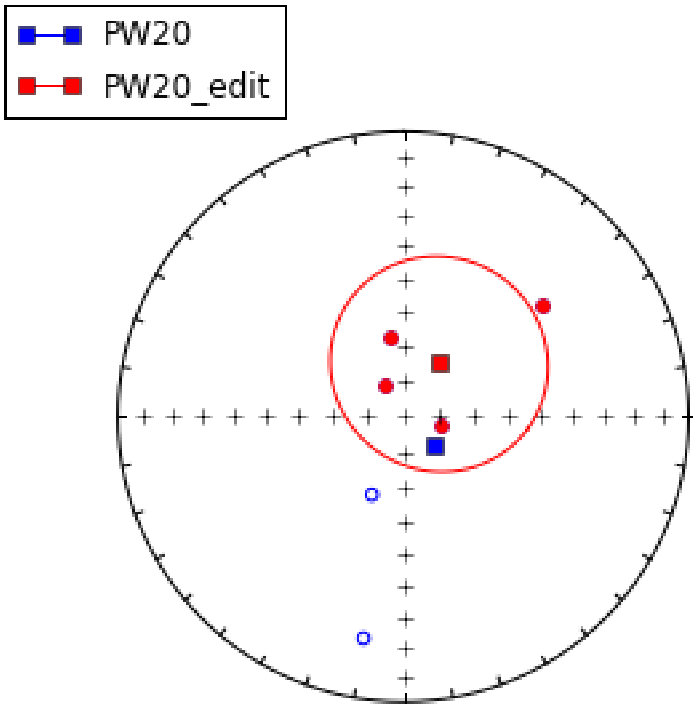
The PW18 data have similar directions to the Karoo paleomagnetic direction. It may be that the Molepolole Prison Quarry Sill is an Umkondo sill with a Karoo overprint. Due to the lack of consistency within the intrusion it is not included in this compilation or the “unknown intrusion” table.

3.4.14 Lentsweletau Sill - PW20

Thermal demag

```
In [61]: PW20_tc_dir=[]
    for n in range(len(PW20_tc)):
        Dec,Inc=PW20_tc['specimen_dec'][n],PW20_tc['specimen_inc'][n]
        PW20_tc_dir.append([Dec,Inc,1.])
    PW20_tc_mean=pmag.fisher_mean(PW20_tc_dir)

    PW20_tc_edit = PW20_tc
```

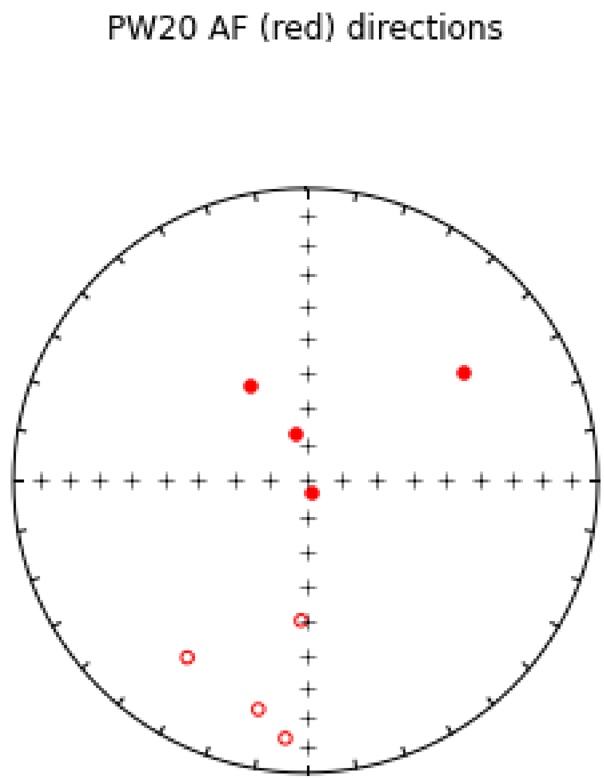
```
PW20_tc_edit = PW20_tc_edit.drop(0)
PW20_tc_edit = PW20_tc_edit.drop(2)
PW20_tc_edit.reset_index(inplace=True)

PW20_tc_edit_dir=[]
for n in range(len(PW20_tc_edit)):
    Dec,Inc=PW20_tc_edit['specimen_dec'][n],PW20_tc_edit['specimen_inc'][n]
    PW20_tc_edit_dir.append([Dec,Inc,1.])
PW20_tc_edit_mean=pmag.fisher_mean(PW20_tc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW20_tc_dir,color='b')
IPmag.iplotDImean(PW20_tc_mean['dec'],PW20_tc_mean['inc'],
                  PW20_tc_mean["alpha95"],color='b',marker='s',label='PW20')
IPmag.iplotDI(PW20_tc_edit_dir,color='r')
IPmag.iplotDImean(PW20_tc_edit_mean['dec'],PW20_tc_edit_mean['inc'],
                  PW20_tc_edit_mean["alpha95"],color='r',marker='s',
                  label='PW20_edit')
```



AF demag

```
In [62]: PW20_AFdata=pd.read_csv('../Data/Botswana_AF/PW20/pmag_specimens.txt',
                                 sep='\t',header=1)
         PW20_AFtc = PW20_AFdata[PW20_AFdata['specimen_tilt_correction'] == 100]
         PW20_AFtc.reset_index(drop=True, inplace=True)

         PW20_AFtc_dir=[]
         for n in range(len(PW20_AFtc)):
             Dec,Inc=PW20_AFtc['specimen_dec'][n],PW20_AFtc['specimen_inc'][n]
             PW20_AFtc_dir.append([Dec,Inc,1.])
         PW20_AFtc_mean=pmag.fisher_mean(PW20_AFtc_dir)

         #Drop redundant specimens from same sample out of AF group.
         PW20_AFtc_edit = PW20_AFtc
```

```
PW20_AFtc_edit.reset_index(drop=True, inplace=True)
PW20_AFtc_edit_dir=[]
for n in range(len(PW20_AFtc_edit)):
    Dec,Inc=PW20_AFtc_edit['specimen_dec'][n],PW20_AFtc_edit['specimen_inc'][n]
    PW20_AFtc_edit_dir.append([Dec,Inc,1.])
PW20_AFtc_edit_mean=pmag.fisher_mean(PW20_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW20_AFtc_edit_dir,color='red')
plt.title('PW20 AF (red) directions')
plt.show()
```

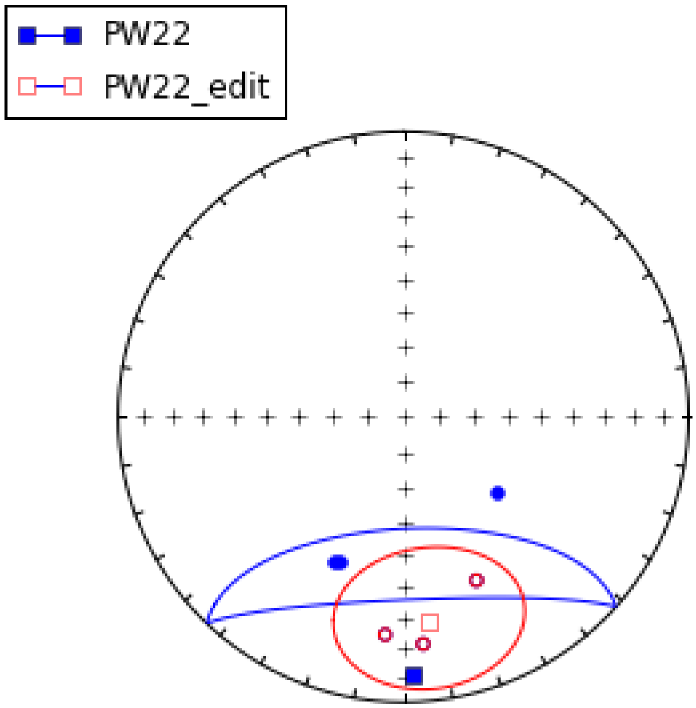


Results for PW20 are scattered and are not considered further.

3.4.15 Mosolotsane 1 Sill - PW21,PW22,JP22,JP23,JP24

Thermal demag (only PW22)

```
In [63]: PW22_tc_dir=[]
         for n in range(len(PW19_tc)):
             Dec,Inc=PW22_tc['specimen_dec'][n],PW22_tc['specimen_inc'][n],
             PW22_tc_dir.append([Dec,Inc,1.])
         PW22_tc_mean=pmag.fisher_mean(PW22_tc_dir)

         PW22_tc_edit = PW22_tc
         PW22_tc_edit = PW22_tc_edit.drop(0)
         PW22_tc_edit = PW22_tc_edit.drop(1)
         PW22_tc_edit = PW22_tc_edit.drop(5)
         PW22_tc_edit.reset_index(inplace=True)

         PW22_tc_edit_dir=[]
         for n in range(len(PW22_tc_edit)):
             Dec,Inc=PW22_tc_edit['specimen_dec'][n],PW22_tc_edit['specimen_inc'][n],
             PW22_tc_edit_dir.append([Dec,Inc,1.])
         PW22_tc_edit_mean=pmag.fisher_mean(PW22_tc_edit_dir)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(PW22_tc_dir,color='b')
         IPmag.iplotDImean(PW22_tc_mean['dec'],PW22_tc_mean['inc'],
                           PW22_tc_mean["alpha95"],color='b',marker='s',label='PW22')
         IPmag.iplotDI(PW22_tc_edit_dir,color='r')
         IPmag.iplotDImean(PW22_tc_edit_mean['dec'],PW22_tc_edit_mean['inc'],
                           PW22_tc_edit_mean["alpha95"],color='r',marker='s',
                           label='PW22_edit')
```

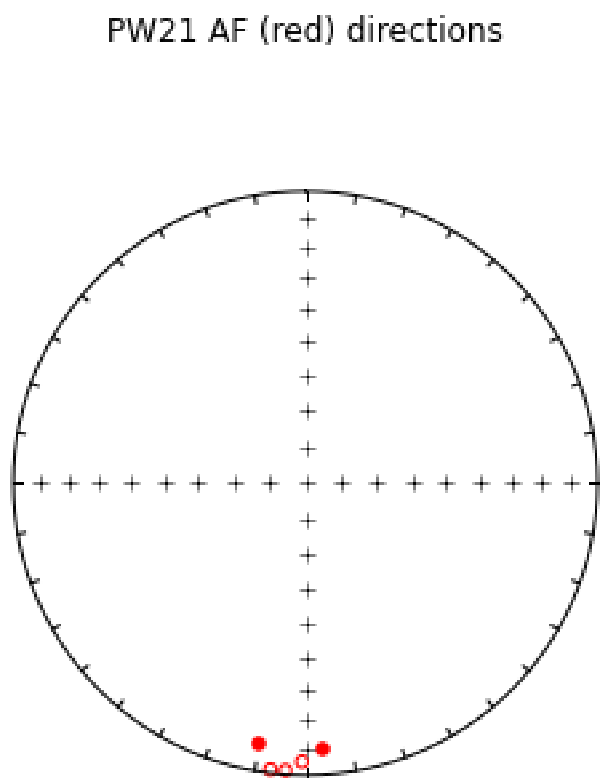


AF demag (PW21 and PW22)

PW21_AF

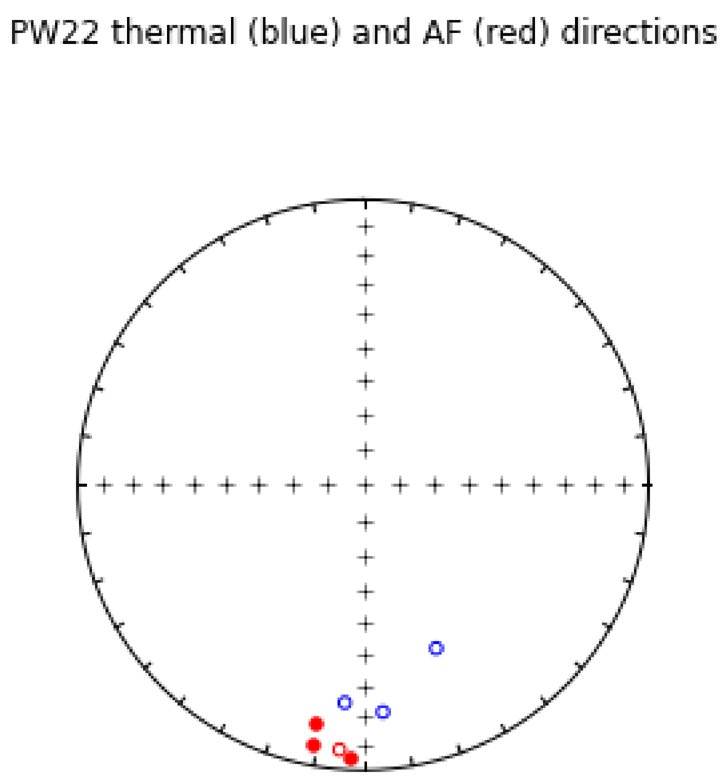
```
In [64]: PW21_AFdata=pd.read_csv('../Data/Botswana_AF/PW21/pmag_specimens.txt',
                                sep='\t',header=1)
     PW21_AFtc = PW21_AFdata[PW21_AFdata['specimen_tilt_correction'] == 100]
     PW21_AFtc.reset_index(drop=True, inplace=True)

     PW21_AFtc_dir=[]
     for n in range(len(PW21_AFtc)):
         Dec,Inc=PW21_AFtc['specimen_dec'][n],PW21_AFtc['specimen_inc'][n]
         PW21_AFtc_dir.append([Dec,Inc,1.])
     PW21_AFtc_mean=pmag.fisher_mean(PW21_AFtc_dir)
```

```
#Drop redundant specimens from same sample out of AF group.
PW21_AFtc_edit = PW21_AFtc
PW21_AFtc_edit.reset_index(drop=True, inplace=True)
PW21_AFtc_edit_dir=[]
for n in range(len(PW21_AFtc_edit)):
    Dec,Inc=PW21_AFtc_edit['specimen_dec'][n],PW21_AFtc_edit['specimen_inc'][n]
    PW21_AFtc_edit_dir.append([Dec,Inc,1.])
PW21_AFtc_edit_mean=pmag.fisher_mean(PW21_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW21_AFtc_edit_dir,color='red')
plt.title('PW21 AF (red) directions')
plt.show()
```



PW22_AF

In [65]:
```
PW22_AFdata=pd.read_csv('../Data/Botswana_AF/PW22/pmag_specimens.txt',
                        sep='\t',header=1)
PW22_AFtc = PW22_AFdata[PW22_AFdata['specimen_tilt_correction'] == 100]
PW22_AFtc.reset_index(drop=True, inplace=True)

PW22_AFtc_dir=[]
for n in range(len(PW22_AFtc)):
    Dec,Inc=PW22_AFtc['specimen_dec'][n],PW22_AFtc['specimen_inc'][n]
    PW22_AFtc_dir.append([Dec,Inc,1.])
PW22_AFtc_mean=pmag.fisher_mean(PW22_AFtc_dir)

#Drop redundant specimens from same sample out of AF group.
PW22_AFtc_edit = PW22_AFtc
PW22_AFtc_edit = PW22_AFtc_edit.drop(3)
PW22_AFtc_edit = PW22_AFtc_edit.drop(4)
PW22_AFtc_edit = PW22_AFtc_edit.drop(5)
PW22_AFtc_edit = PW22_AFtc_edit.drop(0)
PW22_AFtc_edit.reset_index(drop=True, inplace=True)
PW22_AFtc_edit_dir=[]
for n in range(len(PW22_AFtc_edit)):
    Dec,Inc=PW22_AFtc_edit['specimen_dec'][n],PW22_AFtc_edit['specimen_inc'][n]
    PW22_AFtc_edit_dir.append([Dec,Inc,1.])
PW22_AFtc_edit_mean=pmag.fisher_mean(PW22_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW22_tc_edit_dir,color='blue')
IPmag.iplotDI(PW22_AFtc_edit_dir,color='red')
plt.title('PW22 thermal (blue) and AF (red) directions')
plt.show()
```



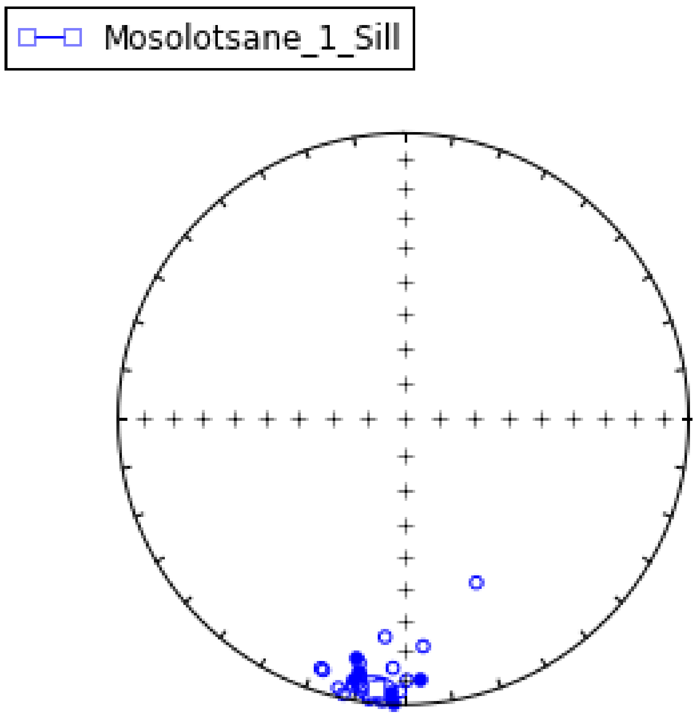
Very good agreement between PW21_AF, PW22_thermal, and PW22_AF. We combine results for a robust mean direction. The Pancake (2001) data from the same intrusion (JP(22,23,24)) is added before calculating a mean.

```
In [66]: JP22_23_24_dir = pickle.load(open('../Data/Pickle/JP22_23_24','rb'))
```

Mosolotsane 1 Sill combined mean - PW21_AF, PW22_ALL, JP(22, 23, 24)

```
In [67]: Mosolotsane_1_Sill=[]
         Mosolotsane_1_Sill = (PW22_tc_edit_dir + PW22_AFtc_edit_dir +
                               PW21_AFtc_edit_dir + JP22_23_24_dir)
         Mosolotsane_1_Sill_mean=pmag.fisher_mean(Mosolotsane_1_Sill)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
```

```
IPmag.iplotNET(1)
IPmag.iplotDI(Mosolotsane_1_Sill,color='blue')
IPmag.iplotDImean(Mosolotsane_1_Sill_mean['dec'],
                  Mosolotsane_1_Sill_mean['inc'],
                  Mosolotsane_1_Sill_mean["alpha95"],color='b',
                  marker='s',label='Mosolotsane_1_Sill')
```



The Pancake data improves the mean greatly and are consistent with our new data and thus are included.

```
In [68]: Intrusion_mean_directions.loc['Mosolotsane_1_Sill']=pd.Series({
             'Intrusion_name':'Mosolotsane_1_Sill',
             'sites_used':'PW21_AF, PW22_ALL, and JP(22,23,24)',
             'site_lat':Site_Locations['LAT(WGS84)'][21],
             'site_long':Site_Locations['LONG(WGS84)'][21],
             'n':int(Mosolotsane_1_Sill_mean['n']),
```

```
            'dec_tc':round(Mosolotsane_1_Sill_mean['dec'],1),
            'inc_tc':round(Mosolotsane_1_Sill_mean['inc'],1),
            'a_95':round(Mosolotsane_1_Sill_mean['alpha95'],1),
            'k':round(Mosolotsane_1_Sill_mean['k'],1),
            'date':'1109.3','date_error':'0.6',
            'dip_direction':262.9,'dip':10})
Intrusion_mean_directions.ix['Mosolotsane_1_Sill']
```

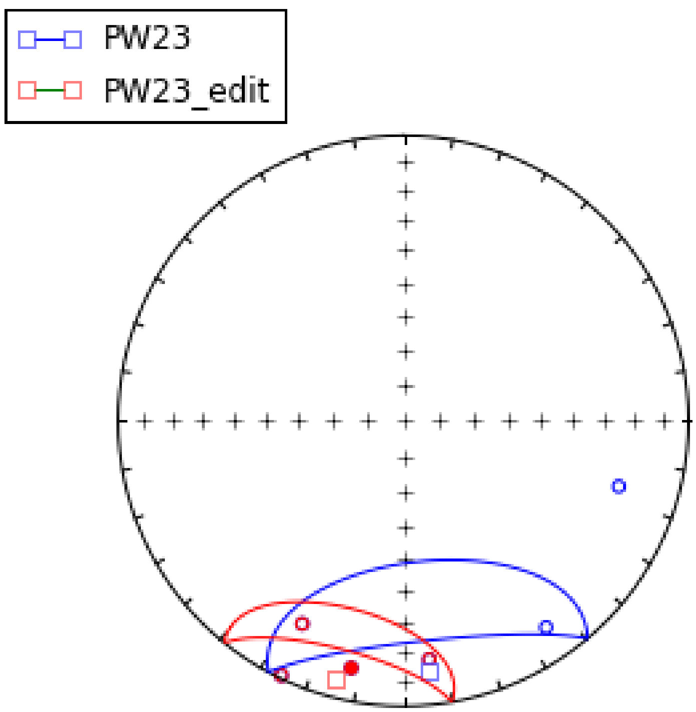
```
Out[68]: Intrusion_name                          Mosolotsane_1_Sill
sites_used         PW21_AF, PW22_ALL, and JP(22,23,24)
site_lat                                      -22.90699
site_long                                      26.38929
n                                                    27
dec_tc                                            186.1
inc_tc                                             -5.6
a_95                                                4.6
k                                                  36.9
date                                             1109.3
date_error                                          0.6
dip_direction                                     262.9
dip                                                  10
Name: Mosolotsane_1_Sill, dtype: object
```

3.4.16 Mosolotsane 5 Sill - PW23

Thermal demag

```
In [69]: PW23_tc_dir=[]
for n in range(len(PW23_tc)):
    Dec,Inc=PW23_tc['specimen_dec'][n],PW23_tc['specimen_inc'][n]
    PW23_tc_dir.append([Dec,Inc,1.])
PW23_tc_mean=pmag.fisher_mean(PW23_tc_dir)

PW23_tc_edit = PW23_tc
PW23_tc_edit = PW23_tc_edit.drop(1)
PW23_tc_edit = PW23_tc_edit.drop(4)
PW23_tc_edit.reset_index(inplace=True)

PW23_tc_edit_dir=[]
for n in range(len(PW23_tc_edit)):
    Dec,Inc=PW23_tc_edit['specimen_dec'][n],PW23_tc_edit['specimen_inc'][n]
    PW23_tc_edit_dir.append([Dec,Inc,1.])
PW23_tc_edit_mean=pmag.fisher_mean(PW23_tc_edit_dir)
```

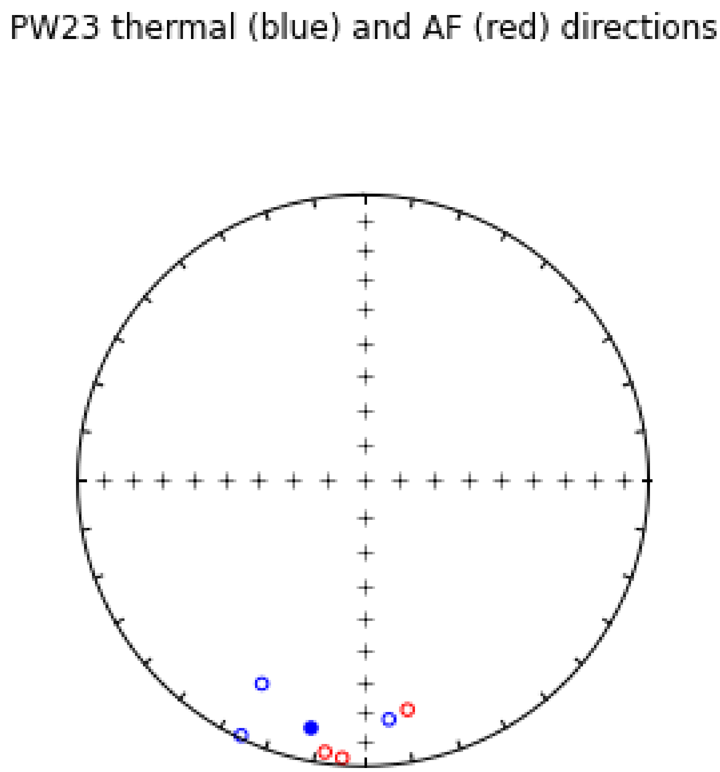
```
fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW23_tc_dir,color='b')
IPmag.iplotDImean(PW23_tc_mean['dec'],PW23_tc_mean['inc'],
                  PW23_tc_mean["alpha95"],color='b',marker='s',label='PW23')
IPmag.iplotDI(PW23_tc_edit_dir,color='r')
IPmag.iplotDImean(PW23_tc_edit_mean['dec'],PW23_tc_edit_mean['inc'],
                  PW23_tc_edit_mean["alpha95"],color='r',marker='s',
                  label='PW23_edit')
```



AF demag

```
In [70]: PW23_AFdata=pd.read_csv('../Data/Botswana_AF/PW23/pmag_specimens.txt',
                                sep='\t',header=1)
```

```
PW23_AFtc = PW23_AFdata[PW23_AFdata['specimen_tilt_correction'] == 100]
PW23_AFtc.reset_index(drop=True, inplace=True)

PW23_AFtc_dir=[]
for n in range(len(PW23_AFtc)):
    Dec,Inc=PW23_AFtc['specimen_dec'][n],PW23_AFtc['specimen_inc'][n]
    PW23_AFtc_dir.append([Dec,Inc,1.])
PW23_AFtc_mean=pmag.fisher_mean(PW23_AFtc_dir)

#Drop redundant specimens from same sample out of AF group.
PW23_AFtc_edit = PW23_AFtc
PW23_AFtc_edit = PW23_AFtc_edit.drop(1)
PW23_AFtc_edit = PW23_AFtc_edit.drop(3)
PW23_AFtc_edit = PW23_AFtc_edit.drop(4)
PW23_AFtc_edit = PW23_AFtc_edit.drop(6)
#Dropped another specimen with anomolously different dec.
PW23_AFtc_edit = PW23_AFtc_edit.drop(5)
PW23_AFtc_edit.reset_index(drop=True, inplace=True)
PW23_AFtc_edit_dir=[]
for n in range(len(PW23_AFtc_edit)):
    Dec,Inc=PW23_AFtc_edit['specimen_dec'][n],PW23_AFtc_edit['specimen_inc'][n]
    PW23_AFtc_edit_dir.append([Dec,Inc,1.])
PW23_AFtc_edit_mean=pmag.fisher_mean(PW23_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW23_tc_edit_dir,color='blue')
IPmag.iplotDI(PW23_AFtc_edit_dir,color='red')
plt.title('PW23 thermal (blue) and AF (red) directions')
plt.show()
```

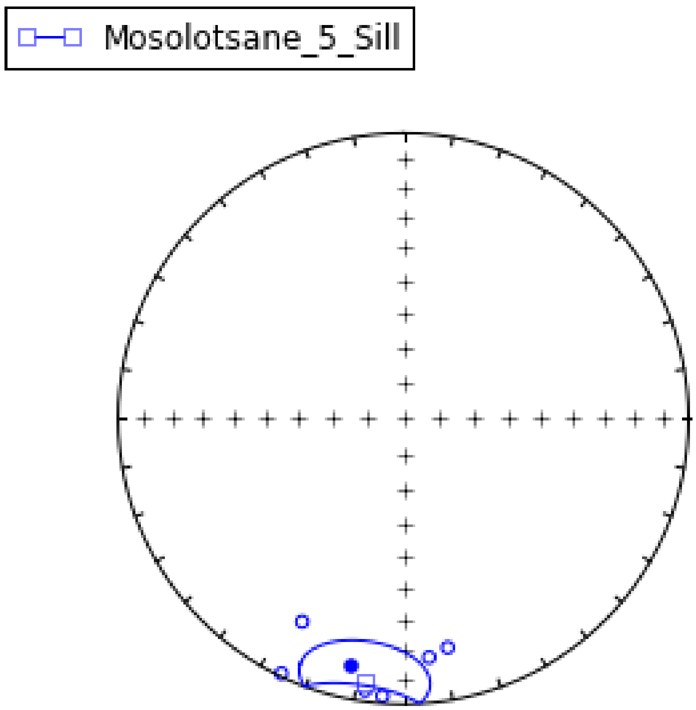


There is very good agreement between all AF and thermal results. Only the 3 AF samples added are shown above.

Mosolotsane 5 Sill combined mean - PW23_ALL Combine the thermal and AF data for an enhanced (and useful) mean.

```
In [71]: Mosolotsane_5_Sill=[]
     Mosolotsane_5_Sill = PW23_tc_edit_dir + PW23_AFtc_edit_dir
     Mosolotsane_5_Sill_mean=pmag.fisher_mean(Mosolotsane_5_Sill)

     fignum = 1
     plt.figure(num=fignum,figsize=(5,5))
     IPmag.iplotNET(1)
     IPmag.iplotDI(Mosolotsane_5_Sill,color='blue')
     IPmag.iplotDImean(Mosolotsane_5_Sill_mean['dec'],
```

```
                Mosolotsane_5_Sill_mean['inc'],
                Mosolotsane_5_Sill_mean["alpha95"],color='b',
                marker='s',label='Mosolotsane_5_Sill')
```




```
In [72]: Intrusion_mean_directions.loc['Mosolotsane_5_Sill'] = pd.Series({
             'Intrusion_name':'Mosolotsane_5_Sill','sites_used':'PW23_ALL',
             'site_lat':Site_Locations['LAT(WGS84)'][22],
             'site_long':Site_Locations['LONG(WGS84)'][22],
             'n':int(Mosolotsane_5_Sill_mean['n']),
             'dec_tc':round(Mosolotsane_5_Sill_mean['dec'],1),
             'inc_tc':round(Mosolotsane_5_Sill_mean['inc'],1),
             'a_95':round(Mosolotsane_5_Sill_mean['alpha95'],1),
             'k':round(Mosolotsane_5_Sill_mean['k'],1),
             'dip_direction':262.9,'dip':10})
     Intrusion_mean_directions.ix['Mosolotsane_5_Sill']
```

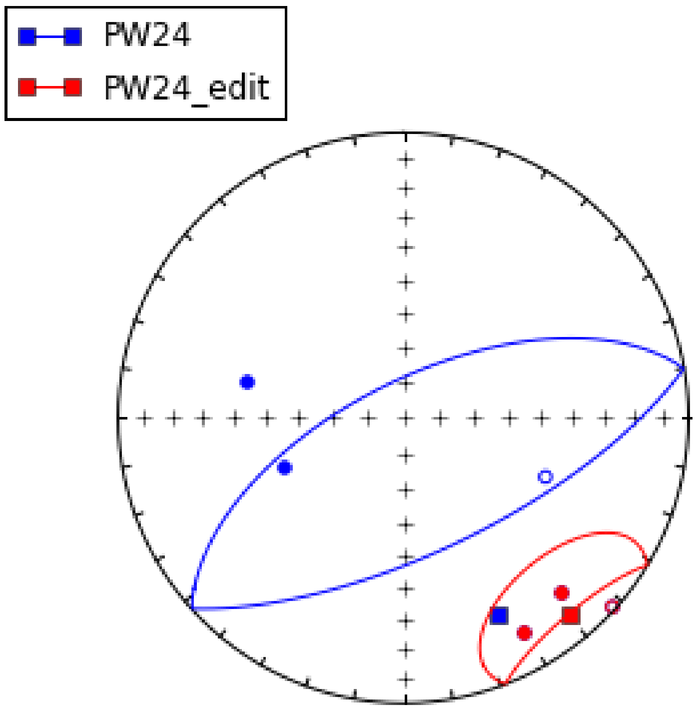
```
Out[72]: Intrusion_name    Mosolotsane_5_Sill
    sites_used                  PW23_ALL
    site_lat                    -22.9033
    site_long                   26.37027
    n                                  7
    dec_tc                         188.5
    inc_tc                          -7.9
    a_95                            14.2
    k                               19.1
    date                             NaN
    date_error                       NaN
    dip_direction                  262.9
    dip                               10
    Name: Mosolotsane_5_Sill, dtype: object
```

3.4.17 Mosolotsane 4 Sill - PW24

Thermal demag

```
In [73]: PW24_tc_dir=[]
    for n in range(len(PW24_tc)):
        Dec,Inc=PW24_tc['specimen_dec'][n],PW24_tc['specimen_inc'][n]
        PW24_tc_dir.append([Dec,Inc,1.])
    PW24_tc_mean=pmag.fisher_mean(PW24_tc_dir)

    PW24_tc_edit = PW24_tc
    PW24_tc_edit = PW24_tc_edit.drop(0)
    PW24_tc_edit = PW24_tc_edit.drop(1)
    PW24_tc_edit = PW24_tc_edit.drop(3)
    PW24_tc_edit.reset_index(inplace=True)

    PW24_tc_edit_dir=[]
    for n in range(len(PW24_tc_edit)):
        Dec,Inc=PW24_tc_edit['specimen_dec'][n],PW24_tc_edit['specimen_inc'][n]
        PW24_tc_edit_dir.append([Dec,Inc,1.])
    PW24_tc_edit_mean=pmag.fisher_mean(PW24_tc_edit_dir)

    fignum = 1
    plt.figure(num=fignum,figsize=(5,5))
    IPmag.iplotNET(1)
    IPmag.iplotDI(PW24_tc_dir,color='b')
    IPmag.iplotDImean(PW24_tc_mean['dec'],PW24_tc_mean['inc'],
                      PW24_tc_mean["alpha95"],color='b',marker='s',label='PW24')
    IPmag.iplotDI(PW24_tc_edit_dir,color='r')
```

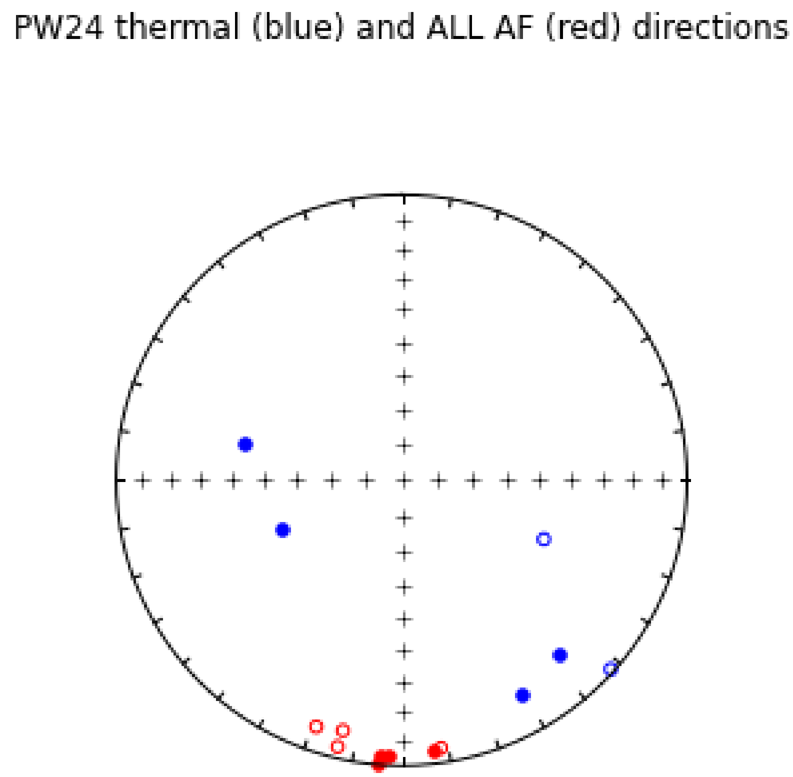
```
IPmag.iplotDImean(PW24_tc_edit_mean['dec'],PW24_tc_edit_mean['inc'],
                  PW24_tc_edit_mean["alpha95"],color='r',marker='s',
                  label='PW24_edit')
```



AF demag

```
In [74]: PW24_AFdata=pd.read_csv('../Data/Botswana_AF/PW24/pmag_specimens.txt',
                                 sep='\t',header=1)
     PW24_AFtc = PW24_AFdata[PW24_AFdata['specimen_tilt_correction'] == 100]
     PW24_AFtc.reset_index(drop=True, inplace=True)

     PW24_AFtc_dir=[]
     for n in range(len(PW24_AFtc)):
         Dec,Inc=PW24_AFtc['specimen_dec'][n],PW24_AFtc['specimen_inc'][n]
```

```
    PW24_AFtc_dir.append([Dec,Inc,1.])
PW24_AFtc_mean=pmag.fisher_mean(PW24_AFtc_dir)

#Drop redundant specimens from same sample out of AF group (none dropped)
PW24_AFtc_edit = PW24_AFtc
PW24_AFtc_edit.reset_index(drop=True, inplace=True)
PW24_AFtc_edit_dir=[]
for n in range(len(PW24_AFtc_edit)):
    Dec,Inc=PW24_AFtc_edit['specimen_dec'][n],PW24_AFtc_edit['specimen_inc'][n]
    PW24_AFtc_edit_dir.append([Dec,Inc,1.])
PW24_AFtc_edit_mean=pmag.fisher_mean(PW24_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW24_tc_dir,color='blue')
IPmag.iplotDI(PW24_AFtc_edit_dir,color='red')
plt.title('PW24 thermal (blue) and ALL AF (red) directions')
plt.show()
```

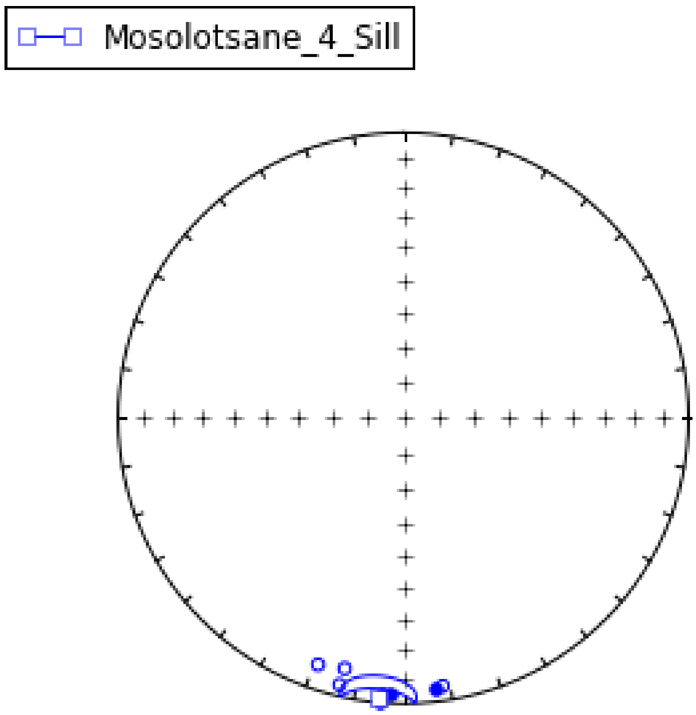


The AF samples are far more consistent than the thermal results. It appears that AF demagnetization was more effective than thermal in isolating a charecteristic remanence for this site. We will use PW24_AF for the mean direction.

Mosolotsane 4 Sill mean - PW24_AF

```
In [75]: Mosolotsane_4_Sill=[]
         Mosolotsane_4_Sill = PW24_AFtc_edit_dir
         Mosolotsane_4_Sill_mean=pmag.fisher_mean(Mosolotsane_4_Sill)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(Mosolotsane_4_Sill,color='blue')
         IPmag.iplotDImean(Mosolotsane_4_Sill_mean['dec'],
```

```
                    Mosolotsane_4_Sill_mean['inc'],
                    Mosolotsane_4_Sill_mean["alpha95"],color='b',
                    marker='s',label='Mosolotsane_4_Sill')
```



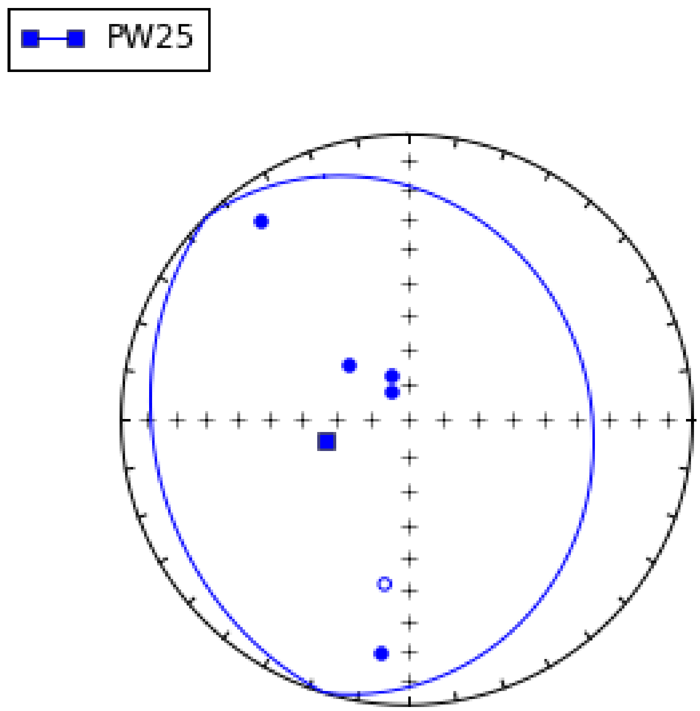

```
In [76]: Intrusion_mean_directions.loc['Mosolotsane_4_Sill']=pd.Series({
             'Intrusion_name':'Mosolotsane_4_Sill',
             'sites_used':'PW24_AF',
             'site_lat':Site_Locations['LAT(WGS84)'][23],
             'site_long':Site_Locations['LONG(WGS84)'][23],
             'n':int(Mosolotsane_4_Sill_mean['n']),
             'dec_tc':round(Mosolotsane_4_Sill_mean['dec'],1),
             'inc_tc':round(Mosolotsane_4_Sill_mean['inc'],1),
             'a_95':round(Mosolotsane_4_Sill_mean['alpha95'],1),
             'k':round(Mosolotsane_4_Sill_mean['k'],1),
             'dip_direction':262.9,'dip':10})
     Intrusion_mean_directions.ix['Mosolotsane_4_Sill']
```

```
Out[76]: Intrusion_name    Mosolotsane_4_Sill
         sites_used                   PW24_AF
         site_lat                   -22.89467
         site_long                    26.3741
         n                                  8
         dec_tc                         185.2
         inc_tc                          -2.5
         a_95                             7.9
         k                               50.3
         date                             NaN
         date_error                       NaN
         dip_direction                  262.9
         dip                               10
         Name: Mosolotsane_4_Sill, dtype: object
```

3.4.18 Mosolotsane 6 Sill - PW25

Thermal demag

```
In [77]: PW25_tc_dir=[]
         for n in range(len(PW25_tc)):
             Dec,Inc=PW25_tc['specimen_dec'][n],PW25_tc['specimen_inc'][n]
             PW25_tc_dir.append([Dec,Inc,1.])
         PW25_tc_mean=pmag.fisher_mean(PW25_tc_dir)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(PW25_tc_dir,color='b')
         IPmag.iplotDImean(PW25_tc_mean['dec'],PW25_tc_mean['inc'],
                           PW25_tc_mean["alpha95"],color='b',marker='s',label='PW25')
```

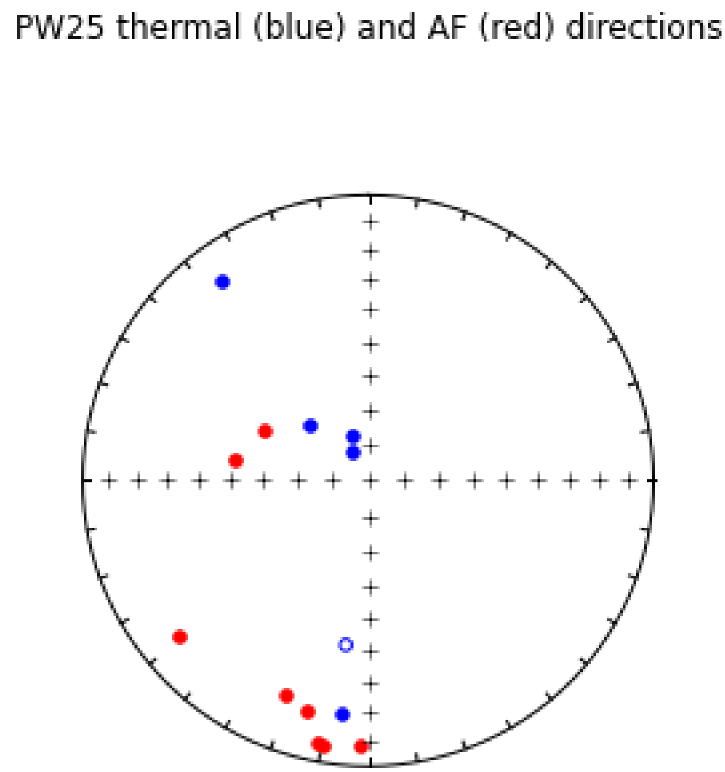


The grouping of three samples that are NW and steeply down could be an overpitn. However, the 5 samples yielded AF results (see below) with directions very similar to other Umkondo intrusions.

AF demag

```
In [78]: PW25_AFdata=pd.read_csv('../Data/Botswana_AF/PW25/pmag_specimens.txt',
                                  sep='\t',header=1)
         PW25_AFtc = PW25_AFdata[PW25_AFdata['specimen_tilt_correction'] == 100]
         PW25_AFtc.reset_index(drop=True, inplace=True)

         PW25_AFtc_dir=[]
         for n in range(len(PW25_AFtc)):
             Dec,Inc=PW25_AFtc['specimen_dec'][n],PW25_AFtc['specimen_inc'][n]
             PW25_AFtc_dir.append([Dec,Inc,1.])
         PW25_AFtc_mean=pmag.fisher_mean(PW25_AFtc_dir)
```

```
#Drop redundant specimens from same sample out of AF group.
PW25_AFtc_edit = PW25_AFtc
PW25_AFtc_edit = PW25_AFtc_edit.drop(0)
PW25_AFtc_edit = PW25_AFtc_edit.drop(3)
PW25_AFtc_edit = PW25_AFtc_edit.drop(7)
PW25_AFtc_edit.reset_index(drop=True, inplace=True)
PW25_AFtc_edit_dir=[]
for n in range(len(PW25_AFtc_edit)):
    Dec,Inc=PW25_AFtc_edit['specimen_dec'][n],PW25_AFtc_edit['specimen_inc'][n]
    PW25_AFtc_edit_dir.append([Dec,Inc,1.])
PW25_AFtc_edit_mean=pmag.fisher_mean(PW25_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW25_tc_dir,color='blue')
IPmag.iplotDI(PW25_AFtc_dir,color='red')
plt.title('PW25 thermal (blue) and AF (red) directions')
plt.show()
```

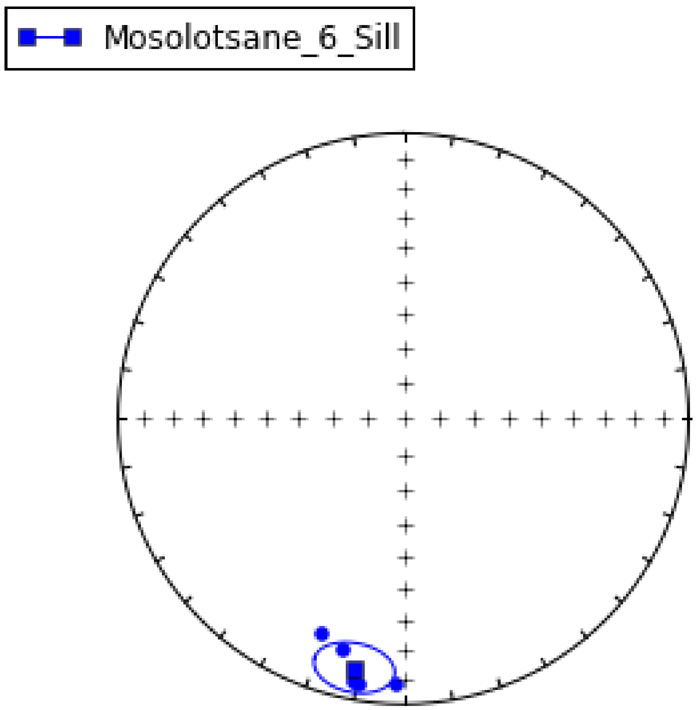


The AF data are much more consistent than the thermal results. AF demagnetization appears to have been more effective in isolating the characteristic remanence than thermal demagnetization. There is only one sample with a thermal result that is in agreement with its AF result. We choose to use the PW25_AF mean for the Mosolotsane 6 Sill.

Mosolotsane 6 Sill mean - PW25_AF

```
In [79]: Mosolotsane_6_Sill=[]
     Mosolotsane_6_Sill = PW25_AFtc_edit_dir
     Mosolotsane_6_Sill_mean=pmag.fisher_mean(Mosolotsane_6_Sill)

     fignum = 1
     plt.figure(num=fignum,figsize=(5,5))
     IPmag.iplotNET(1)
     IPmag.iplotDI(Mosolotsane_6_Sill,color='blue')
```

```
IPmag.iplotDImean(Mosolotsane_6_Sill_mean['dec'],
                  Mosolotsane_6_Sill_mean['inc'],
                  Mosolotsane_6_Sill_mean["alpha95"],color='b',
                  marker='s',label='Mosolotsane_6_Sill')
```



We add the PW25_AF mean direction to the Umkondo cooling unit table.

```
In [80]: Intrusion_mean_directions.loc['Mosolotsane_6_Sill'] = pd.Series({
             'Intrusion_name':'Mosolotsane_6_Sill','sites_used':'PW25_AF',
             'site_lat':Site_Locations['LAT(WGS84)'][24],
             'site_long':Site_Locations['LONG(WGS84)'][24],
             'n':int(Mosolotsane_6_Sill_mean['n']),
             'dec_tc':round(Mosolotsane_6_Sill_mean['dec'],1),
             'inc_tc':round(Mosolotsane_6_Sill_mean['inc'],1),
             'a_95':round(Mosolotsane_6_Sill_mean['alpha95'],1),
             'k':round(Mosolotsane_6_Sill_mean['k'],1),
```

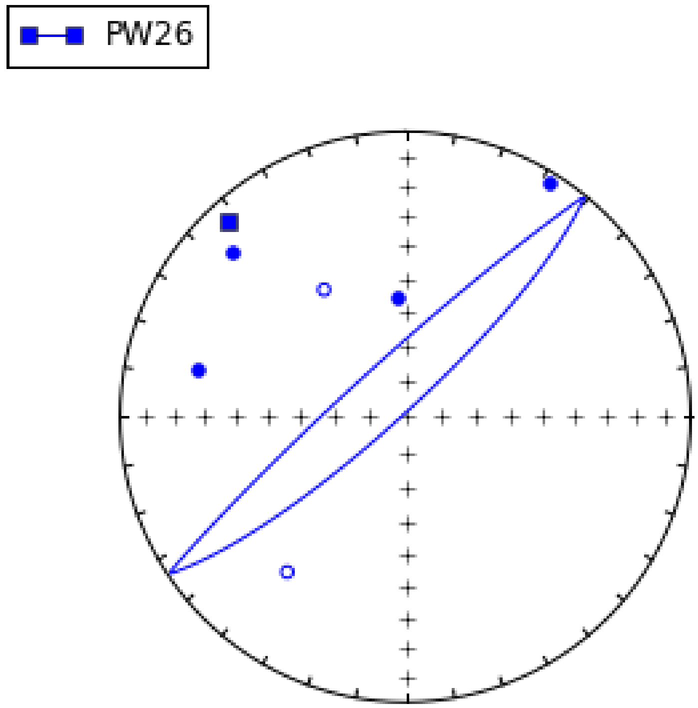
```
            'dip_direction':262.9,'dip':10})
    Intrusion_mean_directions.ix['Mosolotsane_6_Sill']
```

```
Out[80]: Intrusion_name    Mosolotsane_6_Sill
    sites_used                   PW25_AF
    site_lat                    -22.8955
    site_long                   26.36726
    n                                  5
    dec_tc                         191.2
    inc_tc                          11.8
    a_95                               9
    k                               72.7
    date                             NaN
    date_error                       NaN
    dip_direction                  262.9
    dip                               10
    Name: Mosolotsane_6_Sill, dtype: object
```

3.4.19 Mosolotsane 3 Sill - PW26

Thermal demag

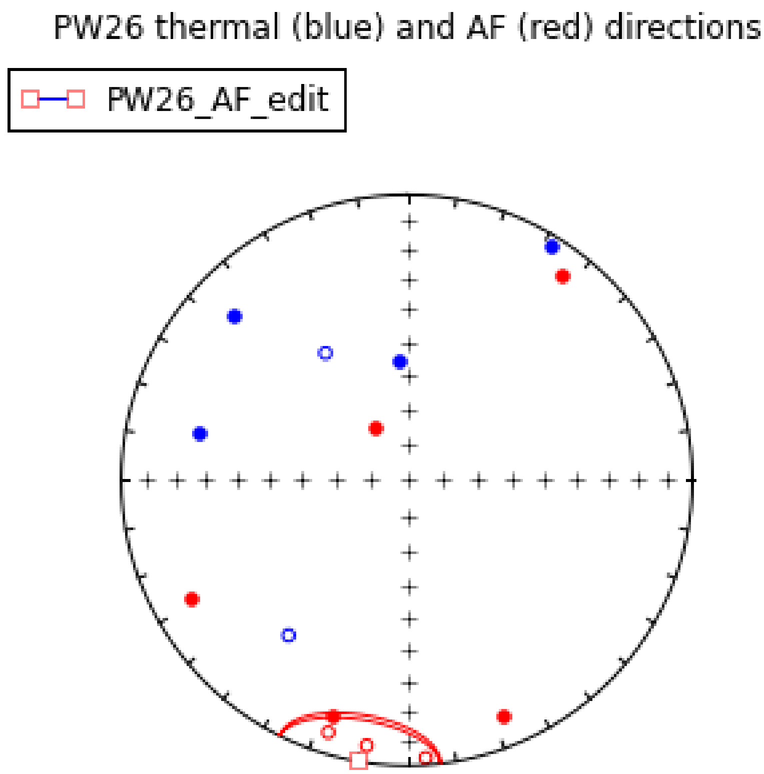
```
In [81]: PW26_tc_dir=[]
    for n in range(len(PW26_tc)):
        Dec,Inc=PW26_tc['specimen_dec'][n],PW26_tc['specimen_inc'][n]
        PW26_tc_dir.append([Dec,Inc,1.])
    PW26_tc_mean=pmag.fisher_mean(PW26_tc_dir)

    #No consistent group, so didn't exclude any samples
    fignum = 1
    plt.figure(num=fignum,figsize=(5,5))
    IPmag.iplotNET(1)
    IPmag.iplotDI(PW26_tc_dir,color='b')
    IPmag.iplotDImean(PW26_tc_mean['dec'],PW26_tc_mean['inc'],
                      PW26_tc_mean["alpha95"],color='b',marker='s',label='PW26')
```



AF demag

```
In [82]: PW26_AFdata=pd.read_csv('../Data/Botswana_AF/PW26/pmag_specimens.txt',
                                  sep='\t',header=1)
         PW26_AFtc = PW26_AFdata[PW26_AFdata['specimen_tilt_correction'] == 100]
         PW26_AFtc.reset_index(drop=True, inplace=True)

         PW26_AFtc_dir=[]
         for n in range(len(PW26_AFtc)):
             Dec,Inc=PW26_AFtc['specimen_dec'][n],PW26_AFtc['specimen_inc'][n]
             PW26_AFtc_dir.append([Dec,Inc,1.])
         PW26_AFtc_mean=pmag.fisher_mean(PW26_AFtc_dir)

         #Drop redundant specimens from same sample out of AF group.
         PW26_AFtc_edit = PW26_AFtc
```

```
PW26_AFtc_edit = PW26_AFtc_edit.drop(0)
PW26_AFtc_edit = PW26_AFtc_edit.drop(1)
PW26_AFtc_edit = PW26_AFtc_edit.drop(5)
PW26_AFtc_edit = PW26_AFtc_edit.drop(6)
PW26_AFtc_edit.reset_index(drop=True, inplace=True)
PW26_AFtc_edit_dir=[]
for n in range(len(PW26_AFtc_edit)):
    Dec,Inc=PW26_AFtc_edit['specimen_dec'][n],PW26_AFtc_edit['specimen_inc'][n]
    PW26_AFtc_edit_dir.append([Dec,Inc,1.])
PW26_AFtc_edit_mean=pmag.fisher_mean(PW26_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW26_tc_dir,color='blue')
IPmag.iplotDI(PW26_AFtc_dir,color='red')
IPmag.iplotDImean(PW26_AFtc_edit_mean['dec'],PW26_AFtc_edit_mean['inc'],
                  PW26_AFtc_edit_mean["alpha95"],color='r',marker='s',
                  label='PW26_AF_edit')
plt.title('PW26 thermal (blue) and AF (red) directions')
plt.show()
```

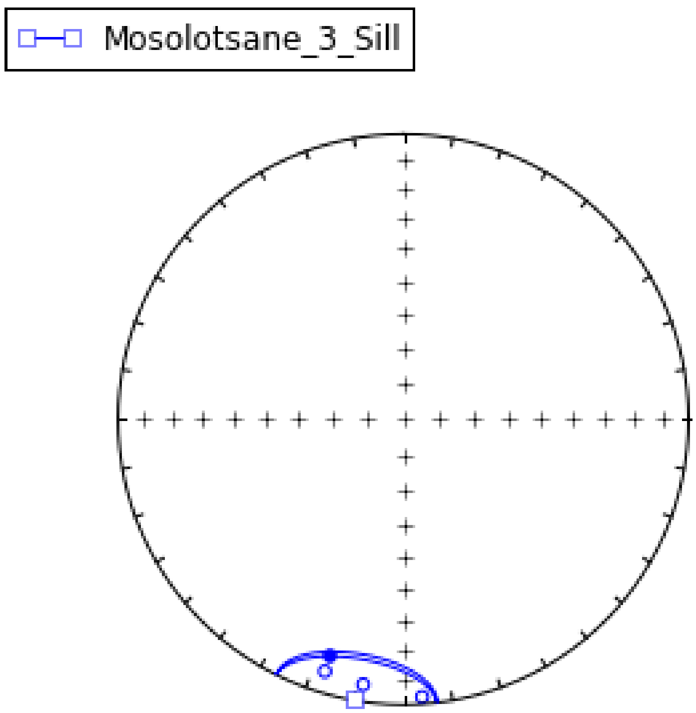


Thermal results are quite scattered. Results from four AF specimens (the same samples were also analyzed thermally) share a consistent Umkondo direction. Although the error of the mean in high, this site will be included in the summary table. The high error on the mean means that it will be filtered out when the grand mean pole is calculated.

Mosolotsane 3 Sill mean - (PW26_AF)

```
In [83]: Mosolotsane_3_Sill=[]
         Mosolotsane_3_Sill = PW26_AFtc_edit_dir
         Mosolotsane_3_Sill_mean=pmag.fisher_mean(Mosolotsane_3_Sill)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(Mosolotsane_3_Sill,color='blue')
```

```
IPmag.iplotDImean(Mosolotsane_3_Sill_mean['dec'],
                  Mosolotsane_3_Sill_mean['inc'],
                  Mosolotsane_3_Sill_mean["alpha95"],color='b',
                  marker='s',label='Mosolotsane_3_Sill')
```



Results from PW26_AF are added to the summary table.

```
In [84]: Intrusion_mean_directions.loc['Mosolotsane_3_Sill'] = pd.Series({
             'Intrusion_name':'Mosolotsane_3_Sill','sites_used':'PW26_AF',
             'site_lat':Site_Locations['LAT(WGS84)'][25],
             'site_long':Site_Locations['LONG(WGS84)'][25],
             'n':int(Mosolotsane_3_Sill_mean['n']),
             'dec_tc':round(Mosolotsane_3_Sill_mean['dec'],1),
             'inc_tc':round(Mosolotsane_3_Sill_mean['inc'],1),
             'a_95':round(Mosolotsane_3_Sill_mean['alpha95'],1),
             'k':round(Mosolotsane_3_Sill_mean['k'],1),
```

```
        'dip_direction':262.9,'dip':10})
    Intrusion_mean_directions.ix['Mosolotsane_3_Sill']
```

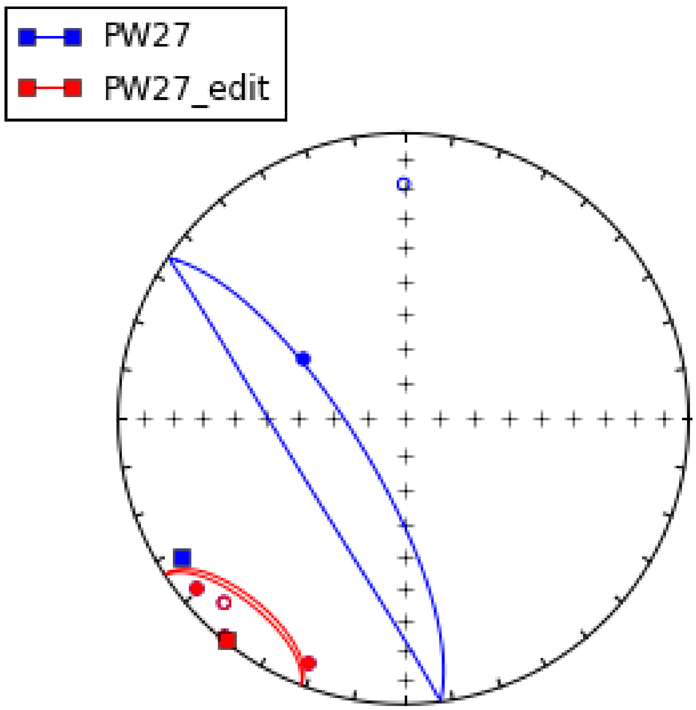
```
Out[84]: Intrusion_name    Mosolotsane_3_Sill
    sites_used                   PW26_AF
    site_lat                   -22.89259
    site_long                   26.38113
    n                                  4
    dec_tc                         189.8
    inc_tc                          -0.9
    a_95                            16.7
    k                               31.3
    date                             NaN
    date_error                       NaN
    dip_direction                  262.9
    dip                               10
    Name: Mosolotsane_3_Sill, dtype: object
```

3.4.20 Mosolotsane 2 Sill - PW27

Thermal demag

```
In [85]: PW27_tc_dir=[]
    for n in range(len(PW27_tc)):
        Dec,Inc=PW27_tc['specimen_dec'][n],PW27_tc['specimen_inc'][n]
        PW27_tc_dir.append([Dec,Inc,1.])
    PW27_tc_mean=pmag.fisher_mean(PW27_tc_dir)


    PW27_tc_edit = PW27_tc
    PW27_tc_edit = PW27_tc_edit.drop(4)
    PW27_tc_edit = PW27_tc_edit.drop(5)
    PW27_tc_edit.reset_index(inplace=True)

    PW27_tc_edit_dir=[]
    for n in range(len(PW27_tc_edit)):
        Dec,Inc=PW27_tc_edit['specimen_dec'][n],PW27_tc_edit['specimen_inc'][n]
        PW27_tc_edit_dir.append([Dec,Inc,1.])
    PW27_tc_edit_mean=pmag.fisher_mean(PW27_tc_edit_dir)


    fignum = 1
    plt.figure(num=fignum,figsize=(5,5))
    IPmag.iplotNET(1)
```

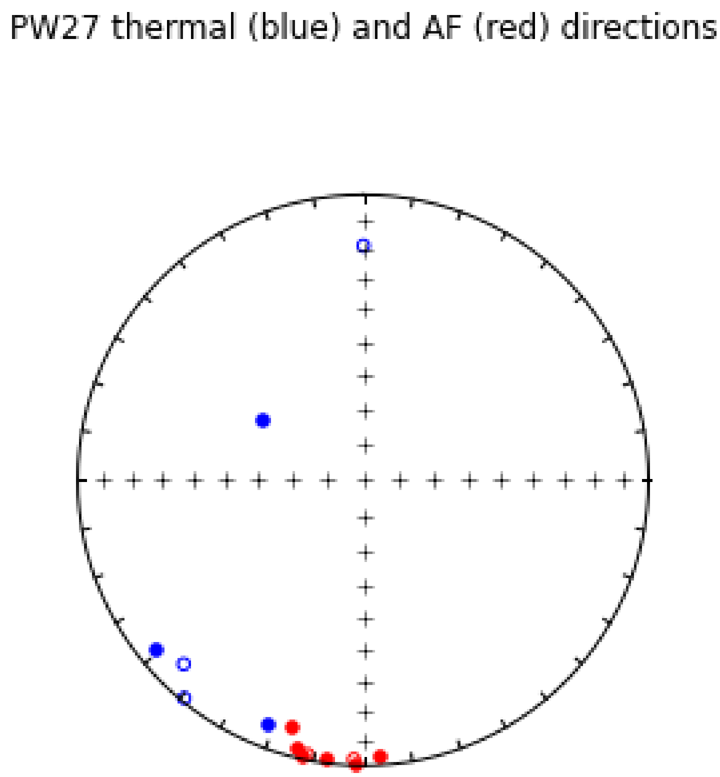
```
IPmag.iplotDI(PW27_tc_dir,color='b')
IPmag.iplotDImean(PW27_tc_mean['dec'],PW27_tc_mean['inc'],
                  PW27_tc_mean["alpha95"],color='b',marker='s',label='PW27')
IPmag.iplotDI(PW27_tc_edit_dir,color='r')
IPmag.iplotDImean(PW27_tc_edit_mean['dec'],PW27_tc_edit_mean['inc'],
                  PW27_tc_edit_mean["alpha95"],color='r',marker='s',
                  label='PW27_edit')
```



AF demag

```
In [86]: PW27_AFdata=pd.read_csv('../Data/Botswana_AF/PW27/pmag_specimens.txt',
                                 sep='\t',header=1)
     PW27_AFtc = PW27_AFdata[PW27_AFdata['specimen_tilt_correction'] == 100]
     PW27_AFtc.reset_index(drop=True, inplace=True)
```

```
PW27_AFtc_dir=[]
for n in range(len(PW27_AFtc)):
    Dec,Inc=PW27_AFtc['specimen_dec'][n],PW27_AFtc['specimen_inc'][n]
    PW27_AFtc_dir.append([Dec,Inc,1.])
PW27_AFtc_mean=pmag.fisher_mean(PW27_AFtc_dir)

#Drop redundant specimens from same sample out of AF group.
PW27_AFtc_edit = PW27_AFtc
PW27_AFtc_edit.reset_index(drop=True, inplace=True)
PW27_AFtc_edit_dir=[]
for n in range(len(PW27_AFtc_edit)):
    Dec,Inc=PW27_AFtc_edit['specimen_dec'][n],PW27_AFtc_edit['specimen_inc'][n]
    PW27_AFtc_edit_dir.append([Dec,Inc,1.])
PW27_AFtc_edit_mean=pmag.fisher_mean(PW27_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW27_tc_dir,color='blue')
IPmag.iplotDI(PW27_AFtc_dir,color='red')
plt.title('PW27 thermal (blue) and AF (red) directions')
plt.show()
```

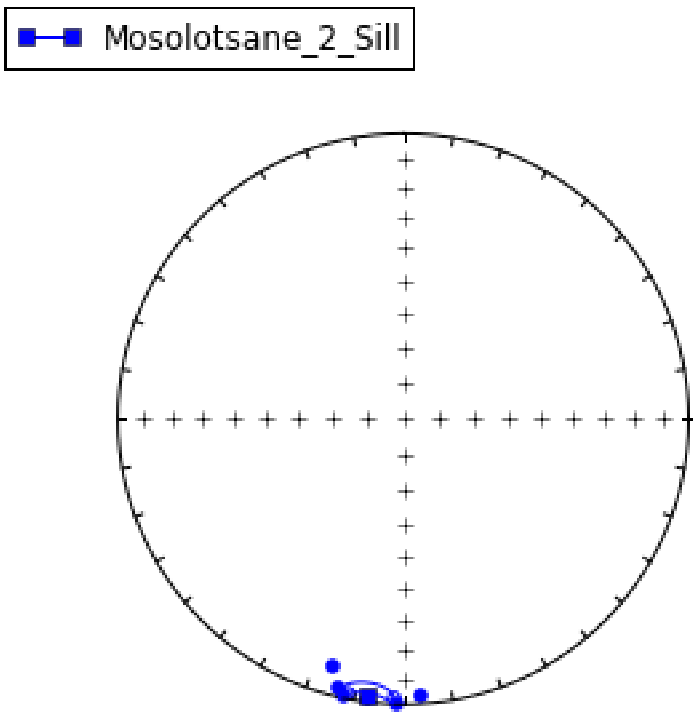


Thermal results are far more scattered than the AF results, although there are some slight similarities in some samples. Due to the consistency of the AF results they are used for the calculation of the mean for the sill and the compilation.

Mosolotsane 2 Sill combined mean (PW27_AF)

```
In [87]: Mosolotsane_2_Sill=[]
    Mosolotsane_2_Sill = PW27_AFtc_edit_dir
    Mosolotsane_2_Sill_mean=pmag.fisher_mean(Mosolotsane_2_Sill)

    fignum = 1
    plt.figure(num=fignum,figsize=(5,5))
    IPmag.iplotNET(1)
    IPmag.iplotDI(Mosolotsane_2_Sill,color='blue')
    IPmag.iplotDImean(Mosolotsane_2_Sill_mean['dec'],
```

```
Mosolotsane_2_Sill_mean['inc'],
Mosolotsane_2_Sill_mean["alpha95"],color='b',
marker='s',label='Mosolotsane_2_Sill')
```




```
In [88]: Intrusion_mean_directions.loc['Mosolotsane_2_Sill'] = pd.Series({
             'Intrusion_name':'Mosolotsane_2_Sill','sites_used':'PW27_AF',
             'site_lat':Site_Locations['LAT(WGS84)'][26],
             'site_long':Site_Locations['LONG(WGS84)'][26],
             'n':int(Mosolotsane_2_Sill_mean['n']),
             'dec_tc':round(Mosolotsane_2_Sill_mean['dec'],1),
             'inc_tc':round(Mosolotsane_2_Sill_mean['inc'],1),
             'a_95':round(Mosolotsane_2_Sill_mean['alpha95'],1),
             'k':round(Mosolotsane_2_Sill_mean['k'],1),
             'dip_direction':262.9,'dip':10})
     Intrusion_mean_directions.ix['Mosolotsane_2_Sill']
```

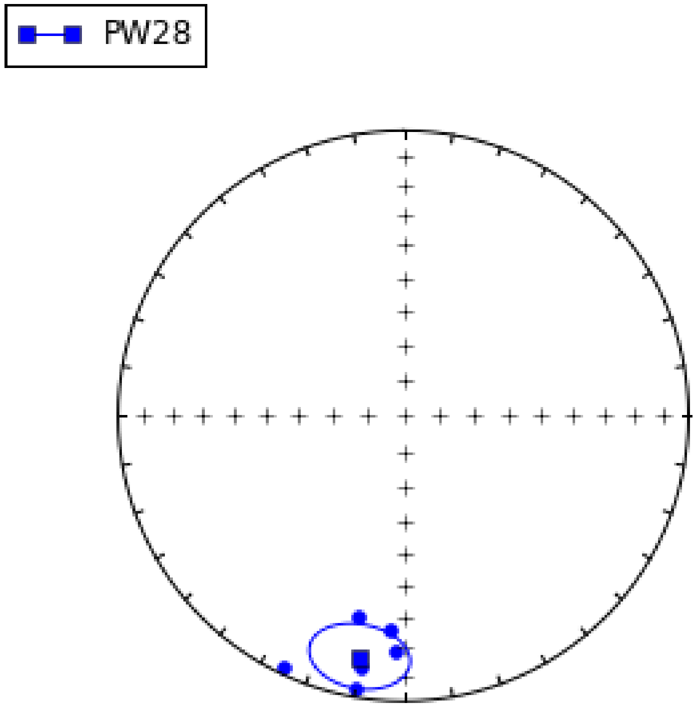
```
Out[88]: Intrusion_name    Mosolotsane_2_Sill
     sites_used                    PW27_AF
     site_lat                    -22.89228
     site_long                    26.38196
     n                                   8
     dec_tc                          187.6
     inc_tc                              2
     a_95                              5.6
     k                                97.5
     date                              NaN
     date_error                        NaN
     dip_direction                   262.9
     dip                                10
     Name: Mosolotsane_2_Sill, dtype: object
```

3.4.21 Shoshong Sill - PW28,JP31,JP33,JP34,J_M(1-6)

All samples are consistent with each other and have great behavior. 'Sites 1 through 6' from Jones and McElhinney (1966) (i.e. J_M(1-6)) were sampled in the Shoshong Sill, but cannot be combined with our new results and those of Pancake because we do not have the complete sample level dataset. The mean we calculate from this sill is a combination of our data and Pancake (2001) data.

Thermal demag

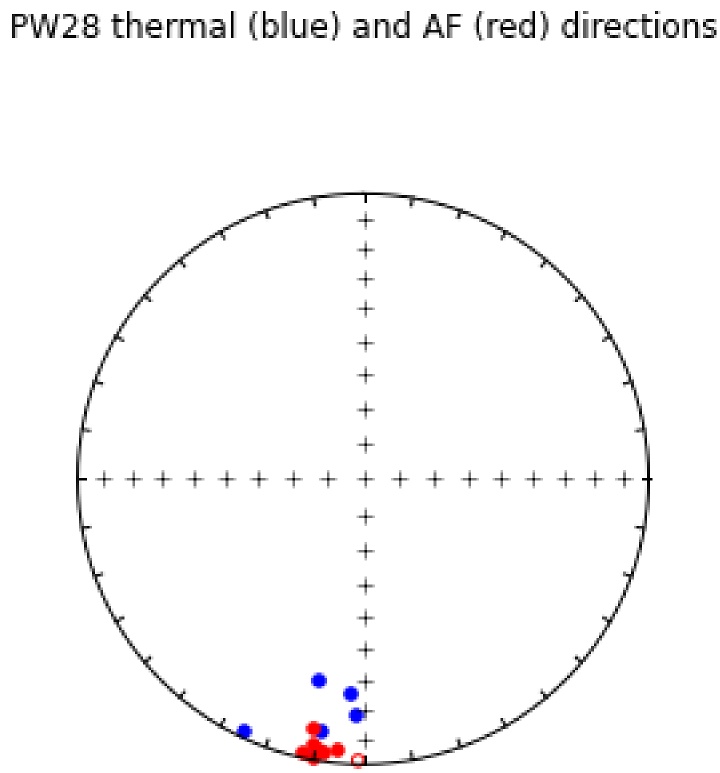
```
In [89]: PW28_tc_dir=[]
     for n in range(len(PW28_tc)):
         Dec,Inc=PW28_tc['specimen_dec'][n],PW28_tc['specimen_inc'][n]
         PW28_tc_dir.append([Dec,Inc,1.])
     PW28_tc_mean=pmag.fisher_mean(PW28_tc_dir)

     fignum = 1
     plt.figure(num=fignum,figsize=(5,5))
     IPmag.iplotNET(1)
     IPmag.iplotDI(PW28_tc_dir,color='b')
     IPmag.iplotDImean(PW28_tc_mean['dec'],PW28_tc_mean['inc'],
                       PW28_tc_mean["alpha95"],color='b',marker='s',label='PW28')
```



AF demag

```
In [90]: PW28_AFdata=pd.read_csv('../Data/Botswana_AF/PW28/pmag_specimens.txt',
                                 sep='\t',header=1)
         PW28_AFtc = PW28_AFdata[PW28_AFdata['specimen_tilt_correction'] == 100]
         PW28_AFtc.reset_index(drop=True, inplace=True)

         PW28_AFtc_dir=[]
         for n in range(len(PW28_AFtc)):
             Dec,Inc=PW28_AFtc['specimen_dec'][n],PW28_AFtc['specimen_inc'][n]
             PW28_AFtc_dir.append([Dec,Inc,1.])
         PW28_AFtc_mean=pmag.fisher_mean(PW28_AFtc_dir)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
```

```
IPmag.iplotNET(1)
IPmag.iplotDI(PW28_tc_dir,color='blue')
IPmag.iplotDI(PW28_AFtc_dir,color='red')
plt.title('PW28 thermal (blue) and AF (red) directions')
plt.show()
```

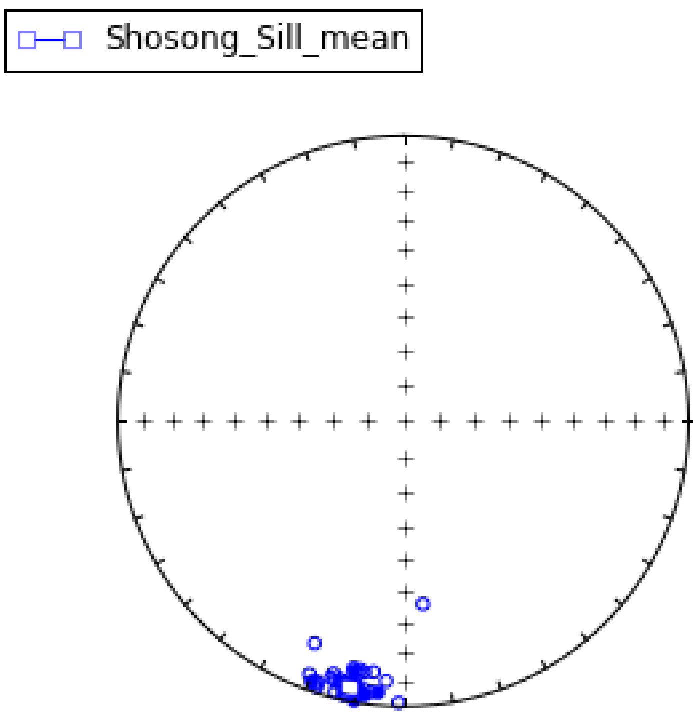


There is good agreement between the thermal and AF results with the AF results having higher precision. We utilize the AF demagnetization data for the mean and also add the data from JP31, JP33, and JP34, which were sampled from the same sill.

```
In [91]: JP26_31_33_34_dir = pickle.load(open('../Data/Pickle/JP26_31_33_34','rb'))
```

Shosong Sill combined mean

```
In [92]: Shoshong_Sill=[]
         Shoshong_Sill = PW28_AFtc_dir + JP26_31_33_34_dir
         Shoshong_Sill_mean=pmag.fisher_mean(Shoshong_Sill)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
         IPmag.iplotNET(1)
         IPmag.iplotDI(Shoshong_Sill,color='blue')
         IPmag.iplotDImean(Shoshong_Sill_mean['dec'],Shoshong_Sill_mean['inc'],
                           Shoshong_Sill_mean["alpha95"],color='b',marker='s',
                           label='Shosong_Sill_mean')
```




```
In [93]: Intrusion_mean_directions.loc['Shoshong_Sill'] = pd.Series({
                 'Intrusion_name':'Shoshong_Sill',
                 'sites_used':'PW28_AF and JP(26,31,33,34)',
```

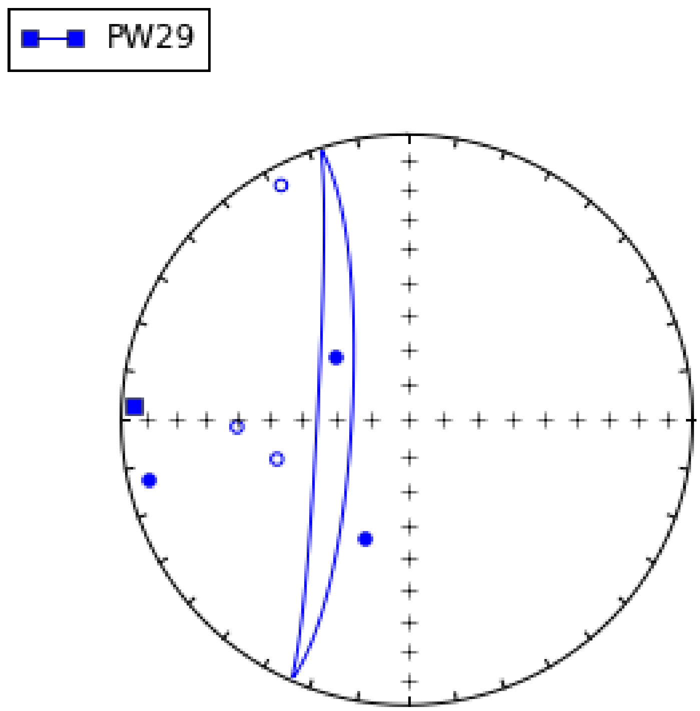
```
            'site_lat':Site_Locations['LAT(WGS84)'][27],
            'site_long':Site_Locations['LONG(WGS84)'][27],
            'n':int(Shoshong_Sill_mean['n']),
            'dec_tc':round(Shoshong_Sill_mean['dec'],1),
            'inc_tc':round(Shoshong_Sill_mean['inc'],1),
            'a_95':round(Shoshong_Sill_mean['alpha95'],1),
            'k':round(Shoshong_Sill_mean['k'],1),
            'date':'1109.3','date_error':'0.4',
            'dip_direction':0,'dip':0})
    Intrusion_mean_directions.ix['Shoshong_Sill']
```

```
Out[93]: Intrusion_name                  Shoshong_Sill
    sites_used        PW28_AF and JP(26,31,33,34)
    site_lat                            -23.00519
    site_long                            26.48383
    n                                          33
    dec_tc                                  191.5
    inc_tc                                   -5.4
    a_95                                      3.1
    k                                        65.2
    date                                   1109.3
    date_error                                0.4
    dip_direction                               0
    dip                                         0
    Name: Shoshong_Sill, dtype: object
```

3.4.22 Phage Sill - PW29

Thermal demag Thermal results are inconsistent.

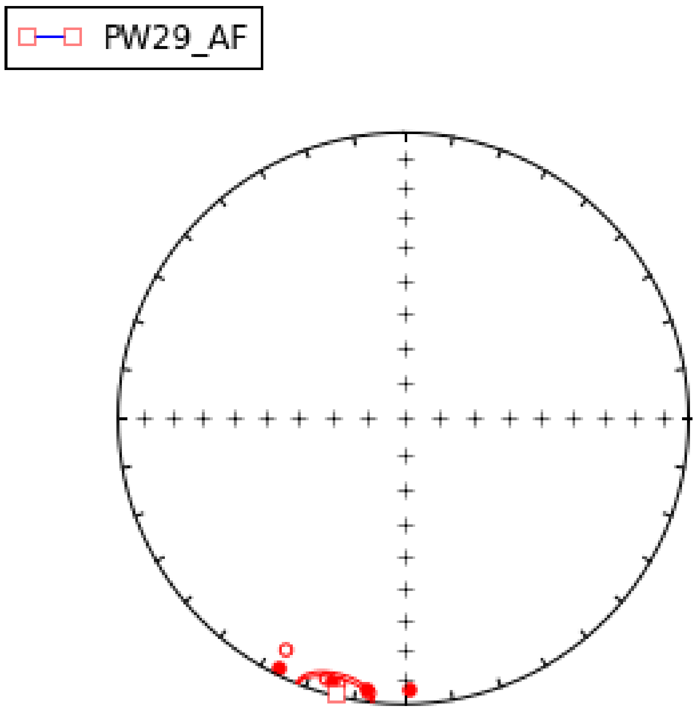
```
In [94]: PW29_tc_dir=[]
    for n in range(len(PW29_tc)):
        Dec,Inc=PW29_tc['specimen_dec'][n],PW29_tc['specimen_inc'][n]
        PW29_tc_dir.append([Dec,Inc,1.])
    PW29_tc_mean=pmag.fisher_mean(PW29_tc_dir)

    fignum = 1
    plt.figure(num=fignum,figsize=(5,5))
    IPmag.iplotNET(1)
    IPmag.iplotDI(PW29_tc_dir,color='b')
    IPmag.iplotDImean(PW29_tc_mean['dec'],PW29_tc_mean['inc'],
                      PW29_tc_mean["alpha95"],color='b',marker='s',
                      label='PW29')
```



AF demag

```
In [95]: PW29_AFdata=pd.read_csv('../Data/Botswana_AF/PW29/pmag_specimens.txt',
                                  sep='\t',header=1)
         PW29_AFtc = PW29_AFdata[PW29_AFdata['specimen_tilt_correction'] == 100]
         PW29_AFtc.reset_index(drop=True, inplace=True)

         PW29_AFtc_dir=[]
         for n in range(len(PW29_AFtc)):
             Dec,Inc=PW29_AFtc['specimen_dec'][n],PW29_AFtc['specimen_inc'][n]
             PW29_AFtc_dir.append([Dec,Inc,1.])
         PW29_AFtc_mean=pmag.fisher_mean(PW29_AFtc_dir)

         fignum = 1
         plt.figure(num=fignum,figsize=(5,5))
```

```
IPmag.iplotNET(1)
IPmag.iplotDI(PW29_AFtc_dir,color='red')
IPmag.iplotDImean(PW29_AFtc_mean['dec'],PW29_AFtc_mean['inc'],
                  PW29_AFtc_mean["alpha95"],color='r',marker='s',
                  label='PW29_AF')
```



The AF demagnetization data are significantly more consisent than the thermally demagnetized specimens. We use the AF mean for this intrusion for the compilation.

```
In [96]: Intrusion_mean_directions.loc['Phage_Sill'] = pd.Series({'Intrusion_name':
                                 'Phage_Sill',
                                 'sites_used':'PW29_AF',
                                 'site_lat':Site_Locations['LAT(WGS84)'][28],
                                 'site_long':Site_Locations['LONG(WGS84)'][28],
                                 'n':int(PW29_AFtc_mean['n']),
                                 'dec_tc':round(PW29_AFtc_mean['dec'],1),
```

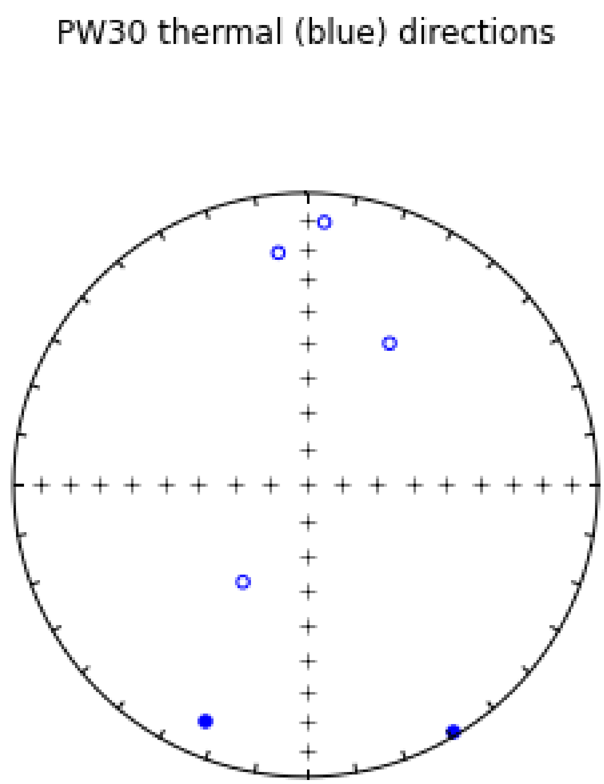
```
                                'inc_tc':round(PW29_AFtc_mean['inc'],1),
                                'a_95':round(PW29_AFtc_mean['alpha95'],1),
                                'k':round(PW29_AFtc_mean['k'],1),
                                'dip_direction':270.0,'dip':10})
Intrusion_mean_directions.ix['Phage_Sill']
```

```
Out[96]: Intrusion_name    Phage_Sill
sites_used           PW29_AF
site_lat           -22.77939
site_long           26.39372
n                          8
dec_tc                   194
inc_tc                  -0.8
a_95                     7.8
k                       50.9
date                     NaN
date_error               NaN
dip_direction            270
dip                       10
Name: Phage_Sill, dtype: object
```

3.4.23 Moijabana Sill - PW30

Thermal demag

```
In [97]: PW30_tc_dir=[]
for n in range(len(PW30_tc)):
    Dec,Inc=PW30_tc['specimen_dec'][n],PW30_tc['specimen_inc'][n]
    PW30_tc_dir.append([Dec,Inc,1.])
PW30_tc_mean=pmag.fisher_mean(PW30_tc_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW30_tc_dir,color='b')
plt.title('PW30 thermal (blue) directions')
plt.show()
```

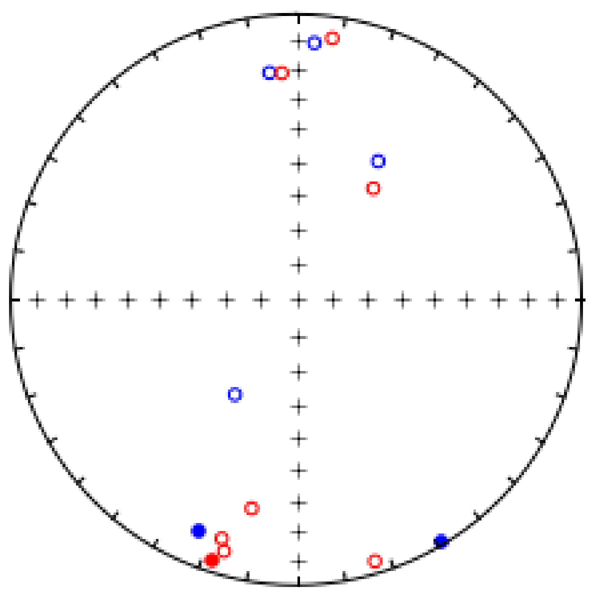


AF demag

```
In [98]: PW30_AFdata=pd.read_csv('../Data/Botswana_AF/PW30/pmag_specimens.txt',
                                 sep='\t',header=1)
         PW30_AFtc = PW30_AFdata[PW30_AFdata['specimen_tilt_correction'] == 100]
         PW30_AFtc.reset_index(drop=True, inplace=True)

         PW30_AFtc_dir=[]
         for n in range(len(PW30_AFtc)):
             Dec,Inc=PW30_AFtc['specimen_dec'][n],PW30_AFtc['specimen_inc'][n]
             PW30_AFtc_dir.append([Dec,Inc,1.])
         PW30_AFtc_mean=pmag.fisher_mean(PW30_AFtc_dir)

         fignum = 1
```

```
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW30_tc_dir,color='blue')
IPmag.iplotDI(PW30_AFtc_dir,color='red')
plt.title('PW28 thermal (blue) and AF (red) directions')
plt.show()
```

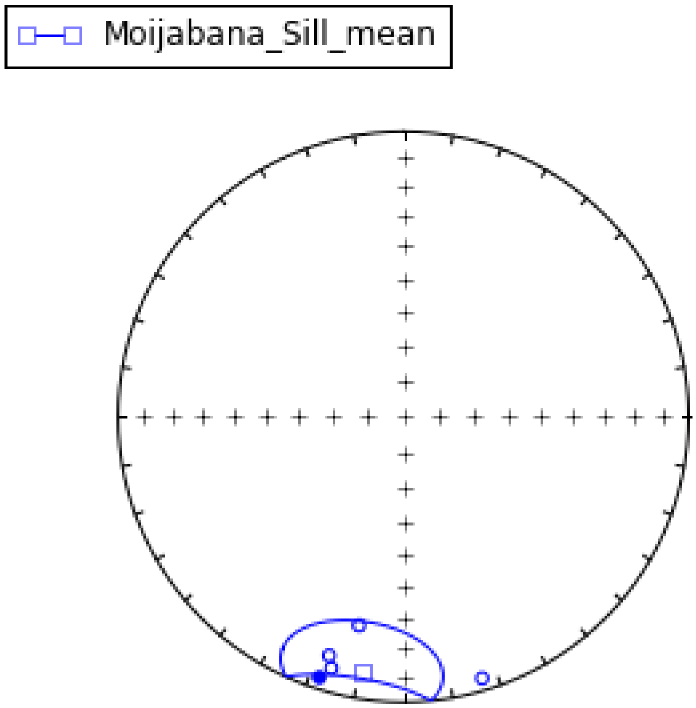
PW28 thermal (blue) and AF (red) directions



The data is rather scattered overall, however there is decent agreement between five samples with an south-directed Umkondo direction. Thermal results near this group are already represented by AF specimens. The three thermal specimens that roughly plot N and up are close to results from AF specimens of the same samples. The edited five sample AF mean will be added to the summary table.

Moijabana Sill mean - PW30_AF

```
In [99]: PW30_AFtc_edit = PW30_AFtc
    PW30_AFtc_edit = PW30_AFtc_edit.drop(3)
    PW30_AFtc_edit = PW30_AFtc_edit.drop(4)
    PW30_AFtc_edit = PW30_AFtc_edit.drop(6)
    PW30_AFtc_edit.reset_index(drop=True, inplace=True)
    PW30_AFtc_edit_dir=[]
    for n in range(len(PW30_AFtc_edit)):
        Dec,Inc=PW30_AFtc_edit['specimen_dec'][n],PW30_AFtc_edit['specimen_inc'][n]
        PW30_AFtc_edit_dir.append([Dec,Inc,1.])
    PW30_AFtc_edit_mean=pmag.fisher_mean(PW30_AFtc_edit_dir)

    Moijabana_Sill=[]
    Moijabana_Sill = PW30_AFtc_edit_dir
    Moijabana_Sill_mean=pmag.fisher_mean(Moijabana_Sill)

    fignum = 1
    plt.figure(num=fignum,figsize=(5,5))
    IPmag.iplotNET(1)
    IPmag.iplotDI(Moijabana_Sill,color='blue')
    IPmag.iplotDImean(Moijabana_Sill_mean['dec'],Moijabana_Sill_mean['inc'],
                      Moijabana_Sill_mean["alpha95"],color='b',marker='s',
                      label='Moijabana_Sill_mean')
```




```
In [100]: Intrusion_mean_directions.loc['Moijabana_Sill'] = pd.Series({'Intrusion_name':
                                   'Moijabana_Sill',
                                   'sites_used':'PW30_AF',
                                   'site_lat':Site_Locations['LAT(WGS84)'][29],
                                   'site_long':Site_Locations['LONG(WGS84)'][29],
                                   'n':int(Moijabana_Sill_mean['n']),
                                   'dec_tc':round(Moijabana_Sill_mean['dec'],1),
                                   'inc_tc':round(Moijabana_Sill_mean['inc'],1),
                                   'a_95':round(Moijabana_Sill_mean['alpha95'],1),
                                   'k':round(Moijabana_Sill_mean['k'],1),
                                   'dip_direction':0,'dip':0})
      Intrusion_mean_directions.ix['Moijabana_Sill']
```

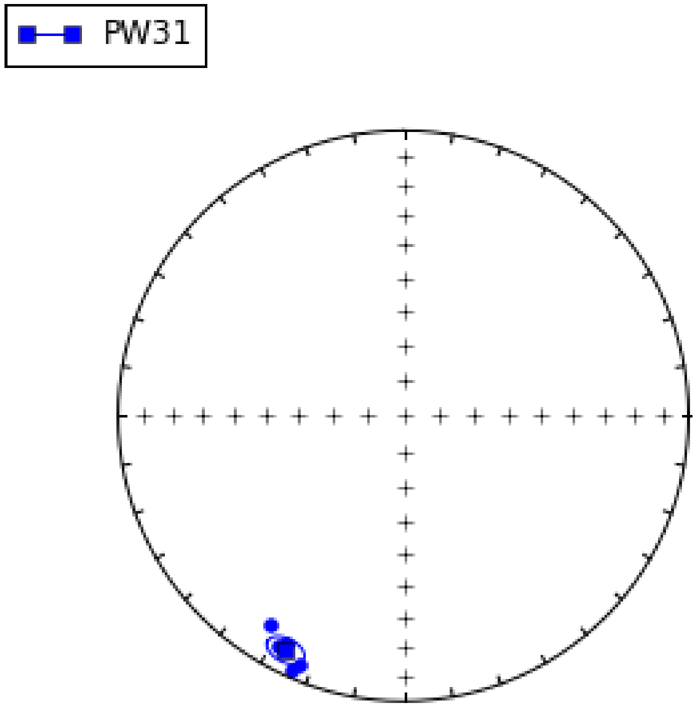
```
Out[100]: Intrusion_name    Moijabana_Sill
      sites_used              PW30_AF
```

```
site_lat                -22.64213
site_long                 26.4426
n                               5
dec_tc                      189.4
inc_tc                        -10
a_95                         17.9
k                            19.2
date                          NaN
date_error                    NaN
dip_direction                   0
dip                             0
Name: Moijabana_Sill, dtype: object
```

3.4.24 Mokgware Sill - PW31, JP30

Thermal demag An excellent site with great agreement between all samples.

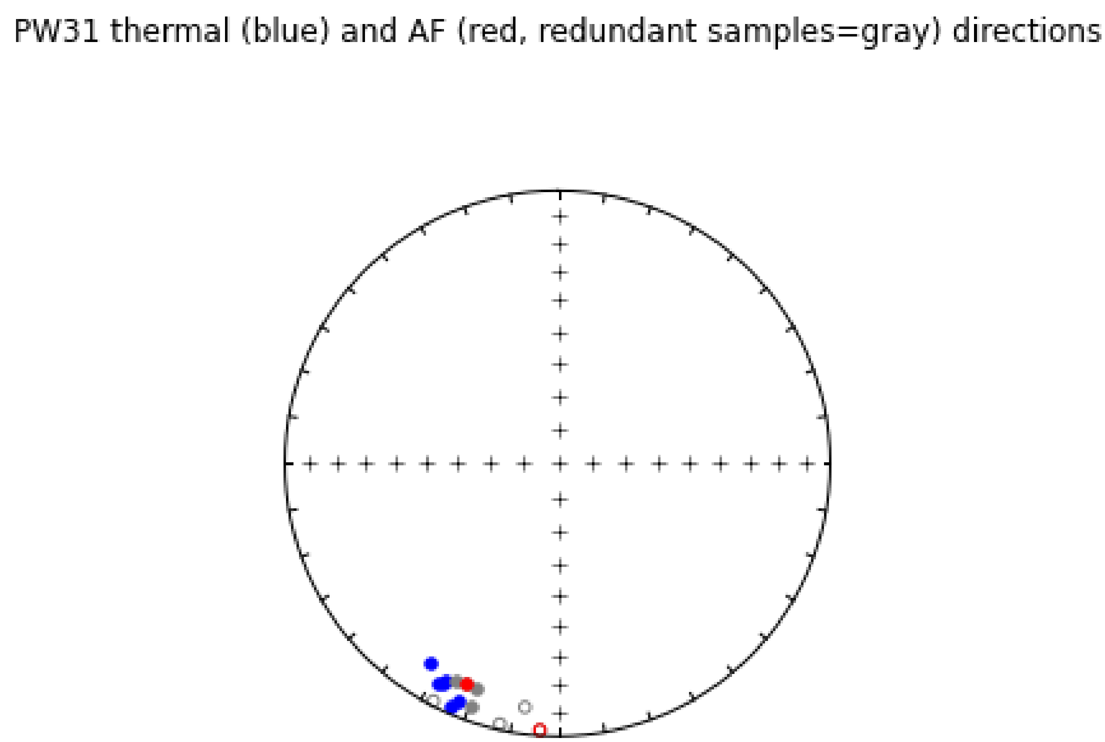
```
In [101]: PW31_tc_dir=[]
     for n in range(len(PW31_tc)):
         Dec,Inc=PW31_tc['specimen_dec'][n],PW31_tc['specimen_inc'][n]
         PW31_tc_dir.append([Dec,Inc,1.])
     PW31_tc_mean=pmag.fisher_mean(PW31_tc_dir)

     fignum = 1
     plt.figure(num=fignum,figsize=(5,5))
     IPmag.iplotNET(1)
     IPmag.iplotDI(PW31_tc_dir,color='b')
     IPmag.iplotDImean(PW31_tc_mean['dec'],PW31_tc_mean['inc'],
                       PW31_tc_mean["alpha95"],color='b',marker='s',
                       label='PW31')
```



AF demag

```
In [102]: PW31_AFdata=pd.read_csv('../Data/Botswana_AF/PW31/pmag_specimens.txt',
                                   sep='\t',header=1)
          PW31_AFtc = PW31_AFdata[PW31_AFdata['specimen_tilt_correction'] == 100]
          PW31_AFtc.reset_index(drop=True, inplace=True)

          PW31_AFtc_dir=[]
          for n in range(len(PW31_AFtc)):
              Dec,Inc=PW31_AFtc['specimen_dec'][n],PW31_AFtc['specimen_inc'][n]
              PW31_AFtc_dir.append([Dec,Inc,1.])
          PW31_AFtc_mean=pmag.fisher_mean(PW31_AFtc_dir)

          #Drop redundant specimens from same sample out of AF group.
          PW31_AFtc_edit = PW31_AFtc
```

```
PW31_AFtc_edit = PW31_AFtc_edit.drop(1)
PW31_AFtc_edit = PW31_AFtc_edit.drop(2)
PW31_AFtc_edit = PW31_AFtc_edit.drop(3)
PW31_AFtc_edit = PW31_AFtc_edit.drop(4)
PW31_AFtc_edit = PW31_AFtc_edit.drop(5)
PW31_AFtc_edit = PW31_AFtc_edit.drop(0)
PW31_AFtc_edit.reset_index(drop=True, inplace=True)
PW31_AFtc_edit_dir=[]
for n in range(len(PW31_AFtc_edit)):
    Dec,Inc=PW31_AFtc_edit['specimen_dec'][n],PW31_AFtc_edit['specimen_inc'][n]
    PW31_AFtc_edit_dir.append([Dec,Inc,1.])
PW31_AFtc_edit_mean=pmag.fisher_mean(PW31_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW31_tc_dir,color='blue')
IPmag.iplotDI(PW31_AFtc_dir,color='gray')
IPmag.iplotDI(PW31_AFtc_edit_dir,color='red')
plt.title('PW31 thermal (blue) and AF (red, redundant samples=gray) directions')
plt.show()
```



The AF data are slightly more scattered with the two non-redundant results plotting near the mean.

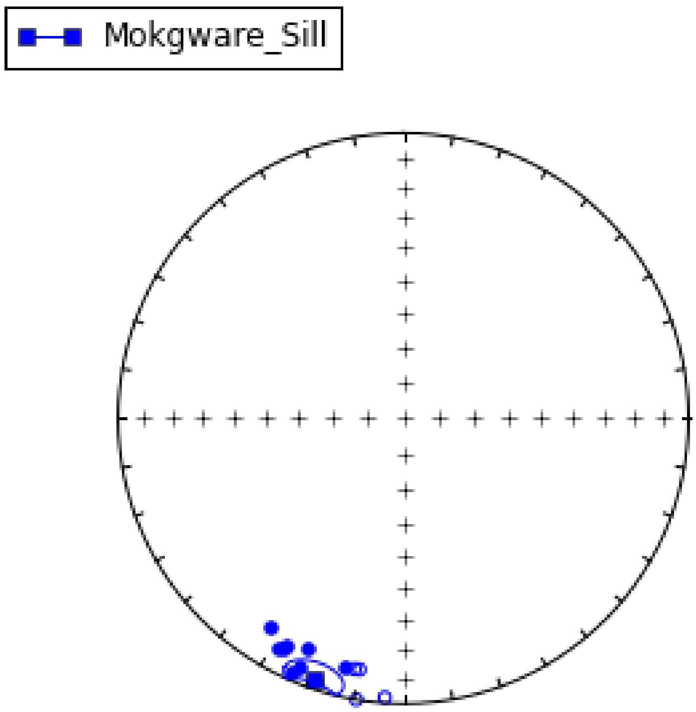
We also need to combine the data from site JP30 with PW31, which was sampled from the same intrusion.

```
In [103]: JP30_dir = pickle.load(open('../Data/Pickle/JP30','rb'))
```

Mokgware Sill combined mean

```
In [104]: Mokgware_Sill=[]
          Mokgware_Sill = PW31_tc_dir + PW31_AFtc_edit_dir + JP30_dir
          Mokgware_Sill_mean=pmag.fisher_mean(Mokgware_Sill)

          fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
```

```
IPmag.iplotNET(1)
IPmag.iplotDI(Mokgware_Sill,color='blue')
IPmag.iplotDImean(Mokgware_Sill_mean['dec'],Mokgware_Sill_mean['inc'],
                  Mokgware_Sill_mean["alpha95"],color='b',marker='s',
                  label='Mokgware_Sill')
```




```
In [105]: Intrusion_mean_directions.loc['Mokgware_Sill'] = pd.Series({
              'Intrusion_name':'Mokgware_Sill','sites_used':'PW31_ALL and JP30',
              'site_lat':Site_Locations['LAT(WGS84)'][30],
              'site_long':Site_Locations['LONG(WGS84)'][30],
              'n':int(Mokgware_Sill_mean['n']),
              'dec_tc':round(Mokgware_Sill_mean['dec'],1),
              'inc_tc':round(Mokgware_Sill_mean['inc'],1),
              'a_95':round(Mokgware_Sill_mean['alpha95'],1),
              'k':round(Mokgware_Sill_mean['k'],1),
```

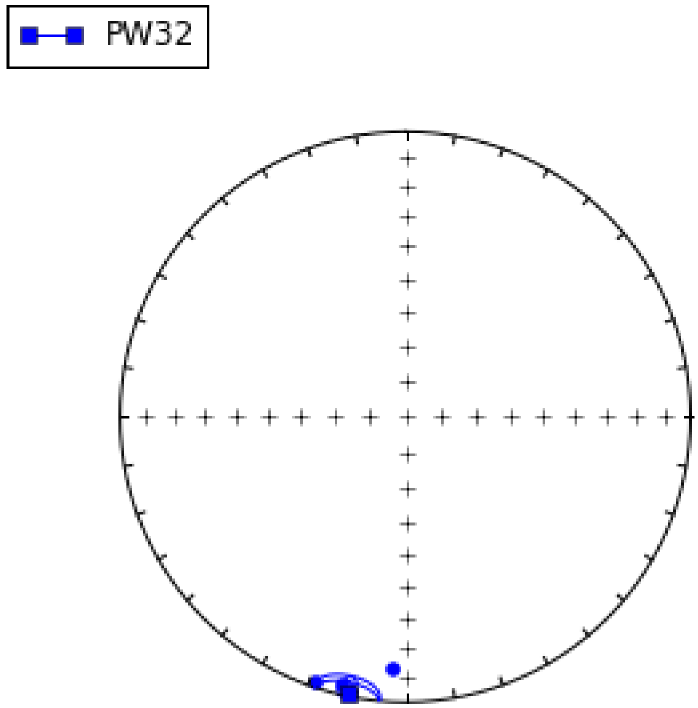
```
          'date':'1112.0','date_error':'0.5',
          'dip_direction':90.1,'dip':4})
Intrusion_mean_directions.ix['Mokgware_Sill']
```

```
Out[105]: Intrusion_name        Mokgware_Sill
sites_used        PW31_ALL and JP30
site_lat                  -22.70685
site_long                  26.61142
n                                13
dec_tc                          199
inc_tc                          3.8
a_95                            6.5
k                              42.2
date                         1112.0
date_error                      0.5
dip_direction                  90.1
dip                               4
Name: Mokgware_Sill, dtype: object
```

3.4.25 Sepatamorire Hill Sill - PW32

Thermal demag Agreement between samples leades to a well-constrained characteristic Umkondo direction.

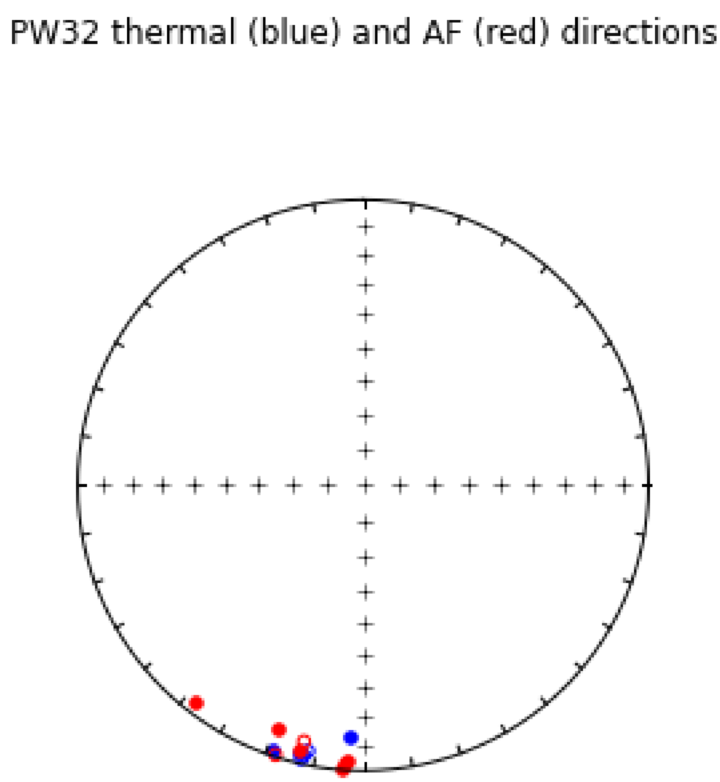
```
In [106]: PW32_tc_dir=[]
for n in range(len(PW32_tc)):
    Dec,Inc=PW32_tc['specimen_dec'][n],PW32_tc['specimen_inc'][n]
    PW32_tc_dir.append([Dec,Inc,1.])
PW32_tc_mean=pmag.fisher_mean(PW32_tc_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW32_tc_dir,color='b')
IPmag.iplotDImean(PW32_tc_mean['dec'],PW32_tc_mean['inc'],
                  PW32_tc_mean["alpha95"],color='b',marker='s',
                  label='PW32')
```



AF demag

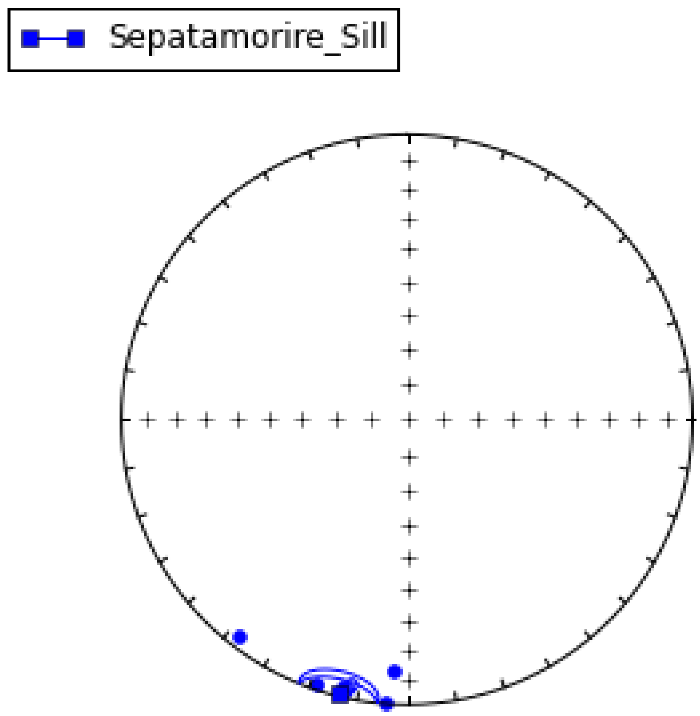
```
In [107]: PW32_AFdata=pd.read_csv('../Data/Botswana_AF/PW32/pmag_specimens.txt',
                                  sep='\t',header=1)
          PW32_AFtc = PW32_AFdata[PW32_AFdata['specimen_tilt_correction'] == 100]
          PW32_AFtc.reset_index(drop=True, inplace=True)

          PW32_AFtc_dir=[]
          for n in range(len(PW32_AFtc)):
              Dec,Inc=PW32_AFtc['specimen_dec'][n],PW32_AFtc['specimen_inc'][n]
              PW32_AFtc_dir.append([Dec,Inc,1.])
          PW32_AFtc_mean=pmag.fisher_mean(PW32_AFtc_dir)

          #Drop redundant specimens from same sample out of AF group.
          PW32_AFtc_edit = PW32_AFtc
```

```
PW32_AFtc_edit = PW32_AFtc_edit.drop(0)
PW32_AFtc_edit = PW32_AFtc_edit.drop(1)
PW32_AFtc_edit = PW32_AFtc_edit.drop(4)
PW32_AFtc_edit = PW32_AFtc_edit.drop(5)
PW32_AFtc_edit = PW32_AFtc_edit.drop(6)
PW32_AFtc_edit = PW32_AFtc_edit.drop(7)
PW32_AFtc_edit.reset_index(drop=True, inplace=True)
PW32_AFtc_edit_dir=[]
for n in range(len(PW32_AFtc_edit)):
    Dec,Inc=PW32_AFtc_edit['specimen_dec'][n],PW32_AFtc_edit['specimen_inc'][n]
    PW32_AFtc_edit_dir.append([Dec,Inc,1.])
PW32_AFtc_edit_mean=pmag.fisher_mean(PW32_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW32_tc_dir,color='blue')
IPmag.iplotDI(PW32_AFtc_dir,color='red')
plt.title('PW32 thermal (blue) and AF (red) directions')
plt.show()
```



Sepatamorire Hill Sill combined mean

```
In [108]: Sepatamorire_Sill=[]
          Sepatamorire_Sill = PW32_tc_dir + PW32_AFtc_edit_dir
          Sepatamorire_Sill_mean=pmag.fisher_mean(Sepatamorire_Sill)

          fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
          IPmag.iplotNET(1)
          IPmag.iplotDI(Sepatamorire_Sill,color='blue')
          IPmag.iplotDImean(Sepatamorire_Sill_mean['dec'],
                            Sepatamorire_Sill_mean['inc'],
                            Sepatamorire_Sill_mean["alpha95"],color='b',
                            marker='s',label='Sepatamorire_Sill')
```



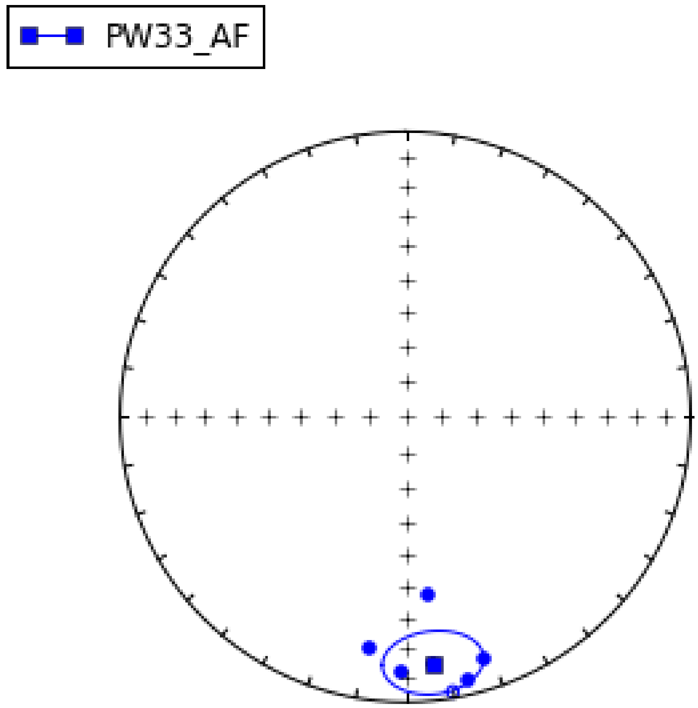

```
In [109]: Intrusion_mean_directions.loc['Sepatamorire_Sill']=pd.Series({
              'Intrusion_name':'Sepatamorire_Sill','sites_used':'PW32_ALL',
              'site_lat':Site_Locations['LAT(WGS84)'][31],
              'site_long':Site_Locations['LONG(WGS84)'][31],
              'n':int(Sepatamorire_Sill_mean['n']),
              'dec_tc':round(Sepatamorire_Sill_mean['dec'],1),
              'inc_tc':round(Sepatamorire_Sill_mean['inc'],1),
              'a_95':round(Sepatamorire_Sill_mean['alpha95'],1),
              'k':round(Sepatamorire_Sill_mean['k'],1),
              'dip_direction':0,'dip':0})
          Intrusion_mean_directions.loc['Sepatamorire_Sill']

Out[109]: Intrusion_name    Sepatamorire_Sill
          sites_used                 PW32_ALL
          site_lat                  -22.33543
```

```
site_long                    26.82285
n                                   8
dec_tc                          194.1
inc_tc                            1.5
a_95                              8.3
k                                45.6
date                              NaN
date_error                        NaN
dip_direction                       0
dip                                 0
Name: Sepatamorire_Sill, dtype: object
```

3.4.26 Palapye Dike - PW33

3.4.27 AF data

```
In [110]: PW33_AFdata=pd.read_csv('../Data/Botswana_AF/PW33/pmag_specimens.txt',
                                  sep='\t',header=1)
      PW33_AFtc = PW33_AFdata[PW33_AFdata['specimen_tilt_correction'] == 100]
      PW33_AFtc.reset_index(drop=True, inplace=True)

      PW33_AFtc_dir=[]
      for n in range(len(PW33_AFtc)):
          Dec,Inc=PW33_AFtc['specimen_dec'][n],PW33_AFtc['specimen_inc'][n]
          PW33_AFtc_dir.append([Dec,Inc,1.])
      PW33_AFtc_mean=pmag.fisher_mean(PW33_AFtc_dir)

      fignum = 1
      plt.figure(num=fignum,figsize=(5,5))
      IPmag.iplotNET(1)
      IPmag.iplotDI(PW33_AFtc_dir,color='blue')
      IPmag.iplotDImean(PW33_AFtc_mean['dec'],PW33_AFtc_mean['inc'],
                        PW33_AFtc_mean["alpha95"],color='blue',marker='s',
                        label='PW33_AF')
```



Palapye Dike mean - PW33

```
In [111]: Palapye_Dike=[]
      Palapye_Dike = PW33_AFtc_dir
      Palapye_Dike_mean=pmag.fisher_mean(Palapye_Dike)

      Intrusion_mean_directions.loc['Palapye_dike'] = pd.Series({'Intrusion_name':
                              'Palapye_dike',
                              'sites_used':'PW33_AF',
                              'site_lat':Site_Locations['LAT(WGS84)'][32],
                              'site_long':Site_Locations['LONG(WGS84)'][32],
                              'n':int(Palapye_Dike_mean['n']),
                              'dec_tc':round(Palapye_Dike_mean['dec'],1),
                              'inc_tc':round(Palapye_Dike_mean['inc'],1),
                              'a_95':round(Palapye_Dike_mean['alpha95'],1),
```

```
                            'k':round(Palapye_Dike_mean['k'],1),
                            'dip_direction':312.1,'dip':9})
Intrusion_mean_directions.ix['Palapye_dike']
```

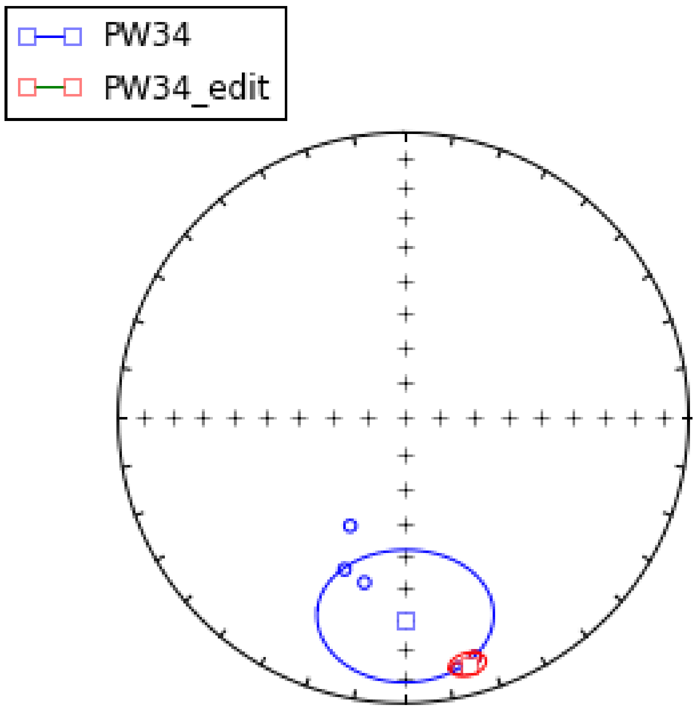
```
Out[111]: Intrusion_name    Palapye_dike
sites_used            PW33_AF
site_lat            -22.57771
site_long            27.28736
n                           7
dec_tc                  173.6
inc_tc                   13.9
a_95                     11.4
k                          29
date                      NaN
date_error                NaN
dip_direction           312.1
dip                         9
Name: Palapye_dike, dtype: object
```

3.4.28 Masama 1 Sill - PW34

Thermal demag

```
In [112]: PW34_tc_dir=[]
for n in range(len(PW34_tc)):
    Dec,Inc=PW34_tc['specimen_dec'][n],PW34_tc['specimen_inc'][n]
    PW34_tc_dir.append([Dec,Inc,1.])
PW34_tc_mean=pmag.fisher_mean(PW34_tc_dir)

PW34_tc_edit = PW34_tc
PW34_tc_edit = PW34_tc_edit.drop(0)
PW34_tc_edit = PW34_tc_edit.drop(1)
PW34_tc_edit = PW34_tc_edit.drop(2)
PW34_tc_edit.reset_index(inplace=True)

PW34_tc_edit_dir=[]
for n in range(len(PW34_tc_edit)):
    Dec,Inc=PW34_tc_edit['specimen_dec'][n],PW34_tc_edit['specimen_inc'][n]
    PW34_tc_edit_dir.append([Dec,Inc,1.])
PW34_tc_edit_mean=pmag.fisher_mean(PW34_tc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
```

```
IPmag.iplotDI(PW34_tc_dir,color='b')
IPmag.iplotDImean(PW34_tc_mean['dec'],PW34_tc_mean['inc'],
                  PW34_tc_mean["alpha95"],color='b',marker='s',
                  label='PW34')
IPmag.iplotDI(PW34_tc_edit_dir,color='r')
IPmag.iplotDImean(PW34_tc_edit_mean['dec'],PW34_tc_edit_mean['inc'],
                  PW34_tc_edit_mean["alpha95"],color='r',marker='s',
                  label='PW34_edit')
```

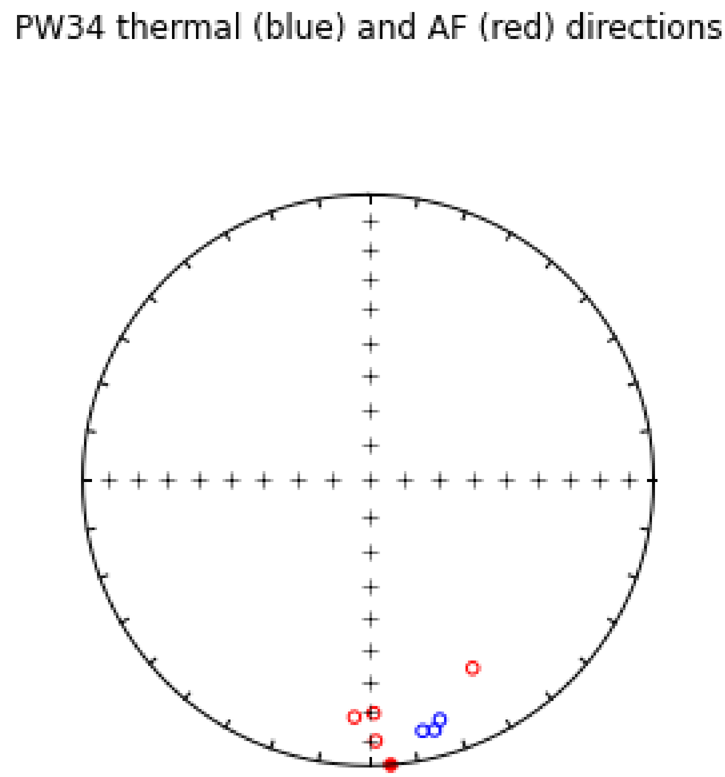


Three of the specimens form a tight grouping.

AF demag

```
In [113]: PW34_AFdata=pd.read_csv('../Data/Botswana_AF/PW34/pmag_specimens.txt',
                                  sep='\t',header=1)
```

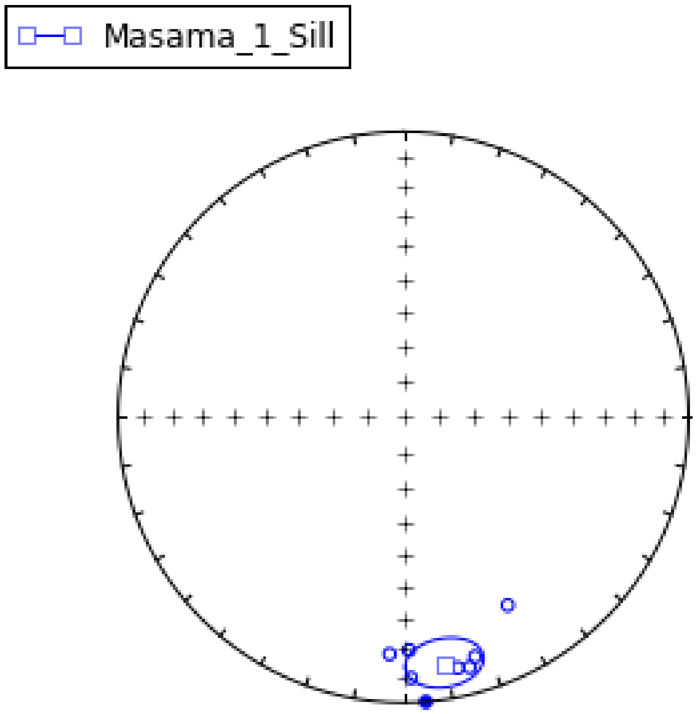
```
PW34_AFtc = PW34_AFdata[PW34_AFdata['specimen_tilt_correction'] == 100]
PW34_AFtc.reset_index(drop=True, inplace=True)

PW34_AFtc_dir=[]
for n in range(len(PW34_AFtc)):
    Dec,Inc=PW34_AFtc['specimen_dec'][n],PW34_AFtc['specimen_inc'][n]
    PW34_AFtc_dir.append([Dec,Inc,1.])
PW34_AFtc_mean=pmag.fisher_mean(PW34_AFtc_dir)

#Drop redundant specimens from same sample out of AF group.
PW34_AFtc_edit = PW34_AFtc
PW34_AFtc_edit = PW34_AFtc_edit.drop(3)
PW34_AFtc_edit = PW34_AFtc_edit.drop(4)
PW34_AFtc_edit = PW34_AFtc_edit.drop(5)
PW34_AFtc_edit.reset_index(drop=True, inplace=True)
PW34_AFtc_edit_dir=[]
for n in range(len(PW34_AFtc_edit)):
    Dec,Inc=PW34_AFtc_edit['specimen_dec'][n],PW34_AFtc_edit['specimen_inc'][n]
    PW34_AFtc_edit_dir.append([Dec,Inc,1.])
PW34_AFtc_edit_mean=pmag.fisher_mean(PW34_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW34_tc_edit_dir,color='blue')
IPmag.iplotDI(PW34_AFtc_edit_dir,color='red')
plt.title('PW34 thermal (blue) and AF (red) directions')
plt.show()
```



The AF results are consistent with the thermal results and contribute to an improved mean.

Masama 1 Sill combined mean

```
In [114]: Masama_1_Sill=[]
      Masama_1_Sill = PW34_tc_edit_dir + PW34_AFtc_edit_dir
      Masama_1_Sill_mean=pmag.fisher_mean(Masama_1_Sill)

      fignum = 1
      plt.figure(num=fignum,figsize=(5,5))
      IPmag.iplotNET(1)
      IPmag.iplotDI(Masama_1_Sill,color='blue')
      IPmag.iplotDImean(Masama_1_Sill_mean['dec'],Masama_1_Sill_mean['inc'],
                        Masama_1_Sill_mean["alpha95"],color='b',marker='s',
                        label='Masama_1_Sill')
```




```
In [115]: Intrusion_mean_directions.loc['Masama_1_Sill'] = pd.Series({
              'Intrusion_name':'Masama_1_Sill','sites_used':'PW34_ALL',
              'site_lat':Site_Locations['LAT(WGS84)'][33],
              'site_long':Site_Locations['LONG(WGS84)'][33],
              'n':int(Masama_1_Sill_mean['n']),
              'dec_tc':round(Masama_1_Sill_mean['dec'],1),
              'inc_tc':round(Masama_1_Sill_mean['inc'],1),
              'a_95':round(Masama_1_Sill_mean['alpha95'],1),
              'k':round(Masama_1_Sill_mean['k'],1),
              'dip_direction':10.4,'dip':8})
      Intrusion_mean_directions.ix['Masama_1_Sill']
```

```
Out[115]: Intrusion_name    Masama_1_Sill
      sites_used         PW34_ALL
      site_lat          -23.81626
```

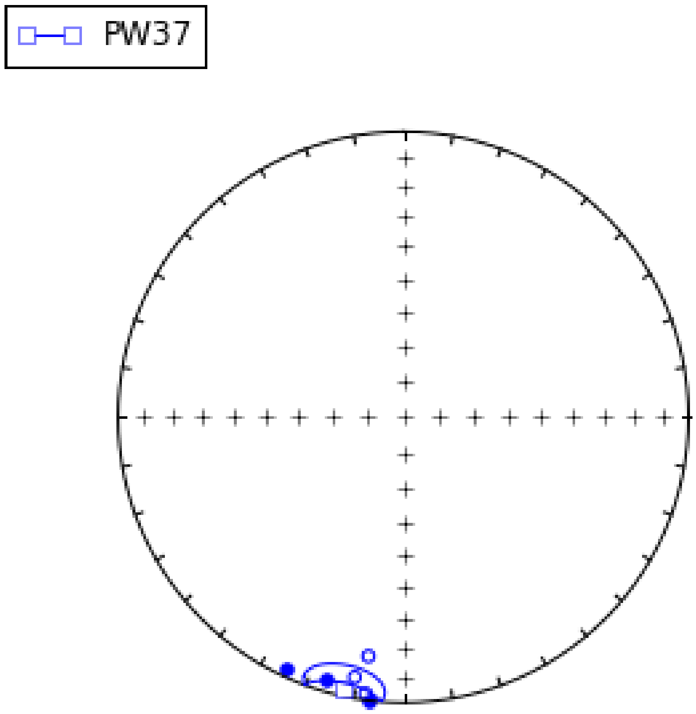
```
site_long               26.73769
n                              8
dec_tc                     170.5
inc_tc                     -13.6
a_95                         8.8
k                           40.3
date                         NaN
date_error                   NaN
dip_direction               10.4
dip                            8
Name: Masama_1_Sill, dtype: object
```

3.4.29 Masama 3 Sill - PW35 and PW37

Only AF demagnetization was conducted on on samples from PW35. Both thermal and AF demagnetization were conducted on samples from PW37.

Thermal demag - PW37

```
In [116]: PW37_tc_dir=[]
    for n in range(len(PW37_tc)):
        Dec,Inc=PW37_tc['specimen_dec'][n],PW37_tc['specimen_inc'][n]
        PW37_tc_dir.append([Dec,Inc,1.])
    PW37_tc_mean=pmag.fisher_mean(PW37_tc_dir)

    fignum = 1
    plt.figure(num=fignum,figsize=(5,5))
    IPmag.iplotNET(1)
    IPmag.iplotDI(PW37_tc_dir,color='b')
    IPmag.iplotDImean(PW37_tc_mean['dec'],PW37_tc_mean['inc'],
                      PW37_tc_mean["alpha95"],color='b',marker='s',
                      label='PW37')
```

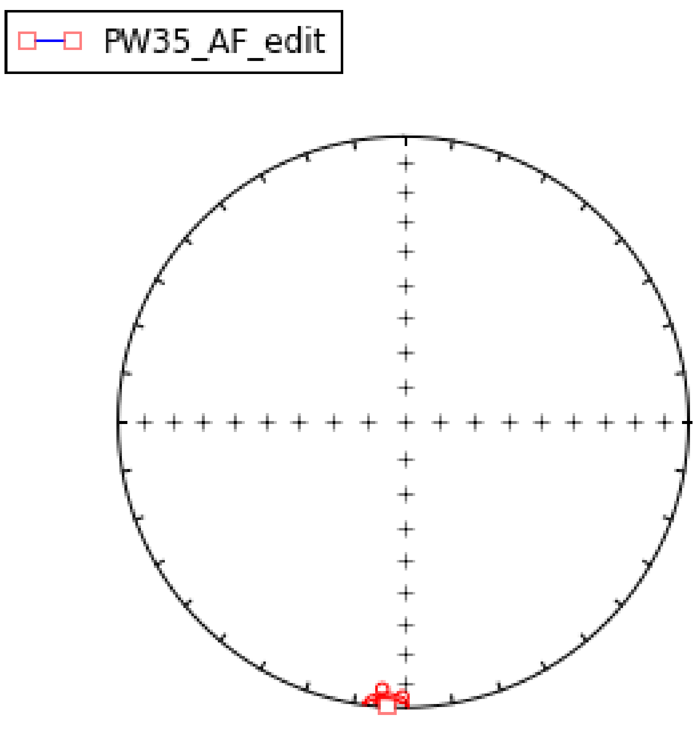


AF demag

PW35_AF

```
In [117]: PW35_AFdata=pd.read_csv('../Data/Botswana_AF/PW35/pmag_specimens.txt',
                                sep='\t',header=1)
          PW35_AFtc = PW35_AFdata[PW35_AFdata['specimen_tilt_correction'] == 100]
          PW35_AFtc.reset_index(drop=True, inplace=True)

          PW35_AFtc_dir=[]
          for n in range(len(PW35_AFtc)):
              Dec,Inc=PW35_AFtc['specimen_dec'][n],PW35_AFtc['specimen_inc'][n]
              PW35_AFtc_dir.append([Dec,Inc,1.])
          PW35_AFtc_mean=pmag.fisher_mean(PW35_AFtc_dir)
```

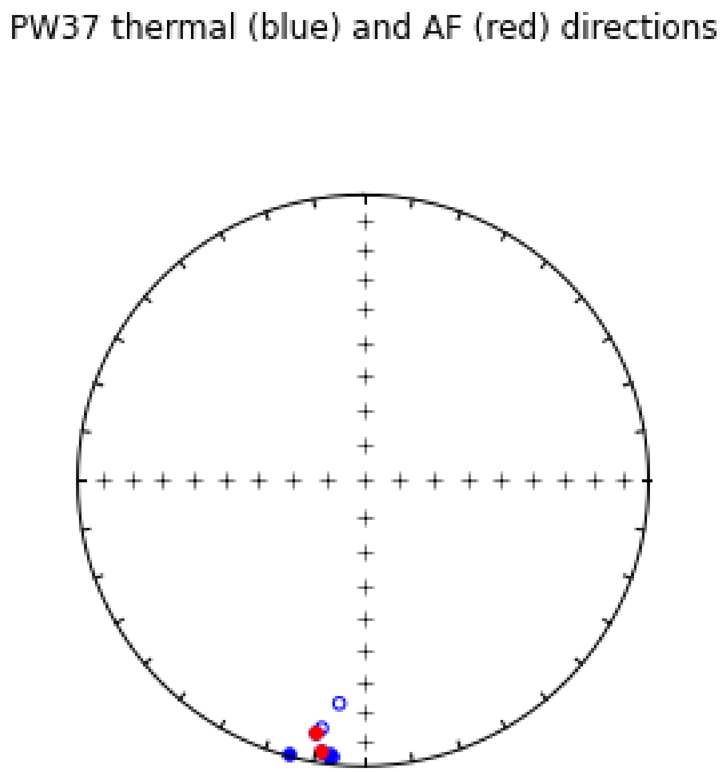
```
#Drop redundant specimens from same sample out of AF group.
PW35_AFtc_edit = PW35_AFtc
PW35_AFtc_edit.reset_index(drop=True, inplace=True)
PW35_AFtc_edit_dir=[]
for n in range(len(PW35_AFtc_edit)):
    Dec,Inc=PW35_AFtc_edit['specimen_dec'][n],PW35_AFtc_edit['specimen_inc'][n]
    PW35_AFtc_edit_dir.append([Dec,Inc,1.])
PW35_AFtc_edit_mean=pmag.fisher_mean(PW35_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW35_AFtc_edit_dir,color='red')
IPmag.iplotDImean(PW35_AFtc_edit_mean['dec'],PW35_AFtc_edit_mean['inc'],
                  PW35_AFtc_edit_mean["alpha95"],color='r',marker='s',
                  label='PW35_AF_edit')
```



Excellent precision on the mean from these data.

PW37_AF

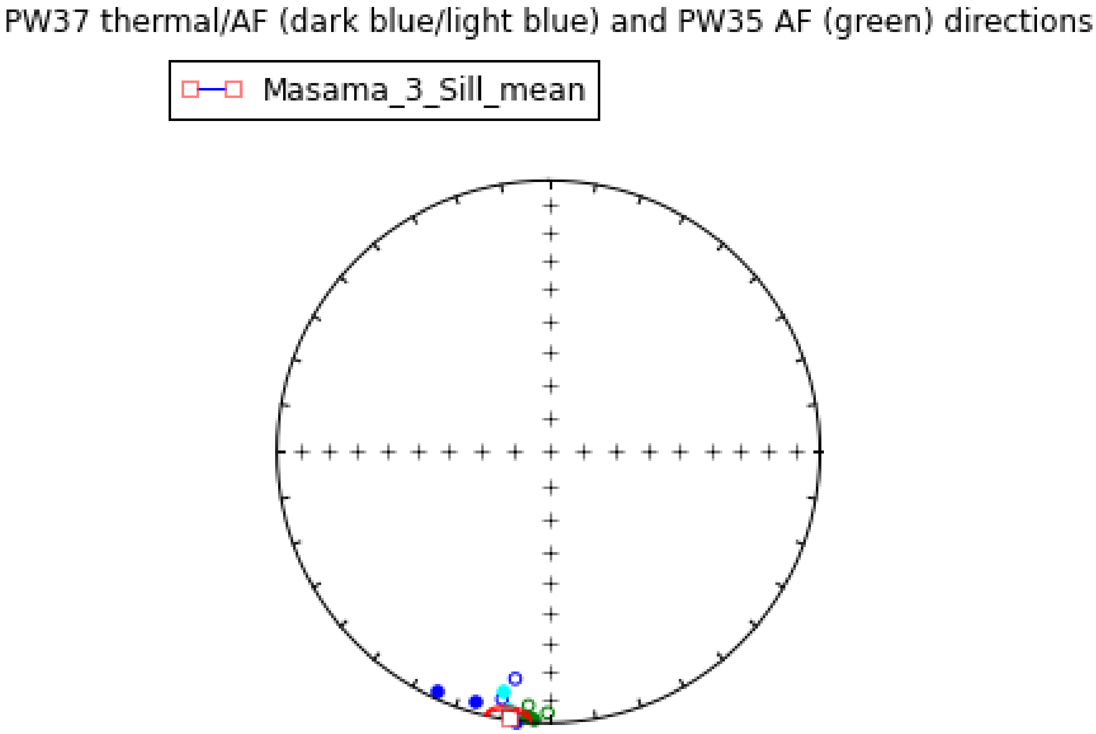
```
In [118]: PW37_AFdata=pd.read_csv('../Data/Botswana_AF/PW37/pmag_specimens.txt',
                                  sep='\t',header=1)
          PW37_AFtc = PW37_AFdata[PW37_AFdata['specimen_tilt_correction'] == 100]
          PW37_AFtc.reset_index(drop=True, inplace=True)

          PW37_AFtc_dir=[]
          for n in range(len(PW37_AFtc)):
              Dec,Inc=PW37_AFtc['specimen_dec'][n],PW37_AFtc['specimen_inc'][n]
              PW37_AFtc_dir.append([Dec,Inc,1.])
          PW37_AFtc_mean=pmag.fisher_mean(PW37_AFtc_dir)

          #Drop redundant specimens from same sample out of AF group.
          PW37_AFtc_edit = PW37_AFtc
          PW37_AFtc_edit = PW37_AFtc_edit.drop(1)
          PW37_AFtc_edit = PW37_AFtc_edit.drop(2)
          PW37_AFtc_edit = PW37_AFtc_edit.drop(3)
          PW37_AFtc_edit = PW37_AFtc_edit.drop(4)
          PW37_AFtc_edit = PW37_AFtc_edit.drop(5)
          PW37_AFtc_edit = PW37_AFtc_edit.drop(6)
          PW37_AFtc_edit.reset_index(drop=True, inplace=True)
          PW37_AFtc_edit_dir=[]
          for n in range(len(PW37_AFtc_edit)):
              Dec,Inc=PW37_AFtc_edit['specimen_dec'][n],PW37_AFtc_edit['specimen_inc'][n]
              PW37_AFtc_edit_dir.append([Dec,Inc,1.])
          PW37_AFtc_edit_mean=pmag.fisher_mean(PW37_AFtc_edit_dir)

          fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
          IPmag.iplotNET(1)
          IPmag.iplotDI(PW37_AFtc_dir,color='blue')
          IPmag.iplotDI(PW37_AFtc_edit_dir,color='red')
          plt.title('PW37 thermal (blue) and AF (red) directions')
          plt.show()
```



Very consistent behavior in samples. Two AF specimen results that do not overlap with thermal specimens are added to the mean.

```
In [119]: Masama_3_Sill=[]
          Masama_3_Sill = PW37_tc_dir + PW37_AFtc_edit_dir + PW35_AFtc_edit_dir
          Masama_3_Sill_mean=pmag.fisher_mean(Masama_3_Sill)

          fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
          IPmag.iplotNET(1)
          IPmag.iplotDI(PW37_tc_dir,color='blue')
          IPmag.iplotDI(PW37_AFtc_edit_dir,color='cyan')
          IPmag.iplotDI(PW35_AFtc_edit_dir,color='green')
          IPmag.iplotDImean(Masama_3_Sill_mean['dec'],Masama_3_Sill_mean['inc'],
                            Masama_3_Sill_mean["alpha95"],color='r',marker='s',
                            label='Masama_3_Sill_mean')
```

```
plt.title('PW37 thermal/AF (dark blue/light blue) and PW35 AF (green) directions')
plt.show()
```




```
In [120]: Intrusion_mean_directions.loc['Masama_3_Sill'] = pd.Series({
              'Intrusion_name':'Masama_3_Sill','sites_used':'PW35_AF and PW37_ALL',
              'site_lat':Site_Locations['LAT(WGS84)'][34],
              'site_long':Site_Locations['LONG(WGS84)'][34],
              'n':int(Masama_3_Sill_mean['n']),
              'dec_tc':round(Masama_3_Sill_mean['dec'],1),
              'inc_tc':round(Masama_3_Sill_mean['inc'],1),
              'a_95':round(Masama_3_Sill_mean['alpha95'],1),
              'k':round(Masama_3_Sill_mean['k'],1),
              'dip_direction':10.4,'dip':8})
          Intrusion_mean_directions.ix['Masama_3_Sill']

Out[120]: Intrusion_name            Masama_3_Sill
```

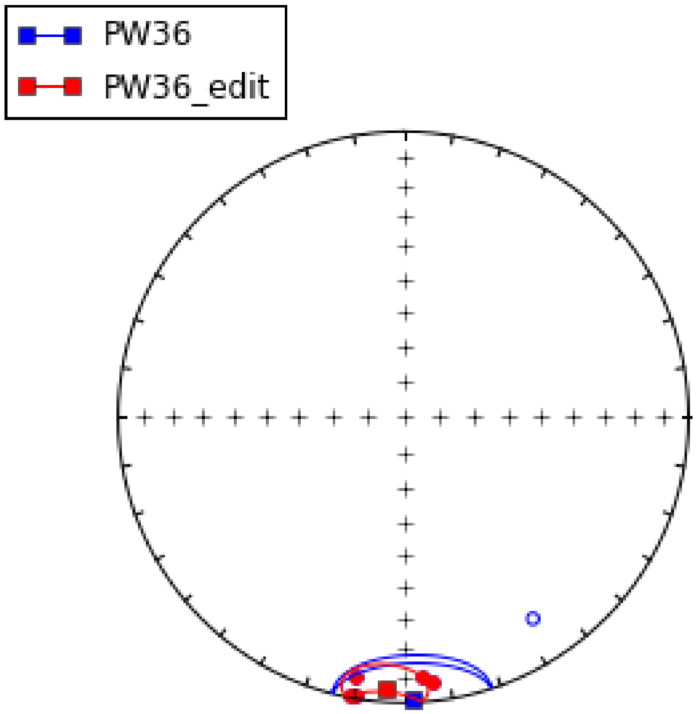
```
sites_used        PW35_AF and PW37_ALL
site_lat                     -23.81403
site_long                     26.73541
n                                   13
dec_tc                           188.5
inc_tc                            -0.7
a_95                               4.8
k                                 75.4
date                               NaN
date_error                         NaN
dip_direction                     10.4
dip                                  8
Name: Masama_3_Sill, dtype: object
```

3.4.30 Masama 2 Sill - PW36

Thermal demag Very well behaved samples that are consistent with each other and an Umkondo direction.

```
In [121]: PW36_tc_dir=[]
          for n in range(len(PW36_tc)):
              Dec,Inc=PW36_tc['specimen_dec'][n],PW36_tc['specimen_inc'][n]
              PW36_tc_dir.append([Dec,Inc,1.])
          PW36_tc_mean=pmag.fisher_mean(PW36_tc_dir)

          PW36_tc_edit = PW36_tc
          PW36_tc_edit = PW36_tc_edit.drop(5)
          PW36_tc_edit.reset_index(inplace=True)

          PW36_tc_edit_dir=[]
          for n in range(len(PW36_tc_edit)):
              Dec,Inc=PW36_tc_edit['specimen_dec'][n],PW36_tc_edit['specimen_inc'][n]
              PW36_tc_edit_dir.append([Dec,Inc,1.])
          PW36_tc_edit_mean=pmag.fisher_mean(PW36_tc_edit_dir)

          fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
          IPmag.iplotNET(1)
          IPmag.iplotDI(PW36_tc_dir,color='b')
          IPmag.iplotDImean(PW36_tc_mean['dec'],PW36_tc_mean['inc'],
                            PW36_tc_mean["alpha95"],color='b',marker='s',
                            label='PW36')
          IPmag.iplotDI(PW36_tc_edit_dir,color='r')
          IPmag.iplotDImean(PW36_tc_edit_mean['dec'],PW36_tc_edit_mean['inc'],
```

```
PW36_tc_edit_mean["alpha95"],color='r',marker='s',
label='PW36_edit')
```

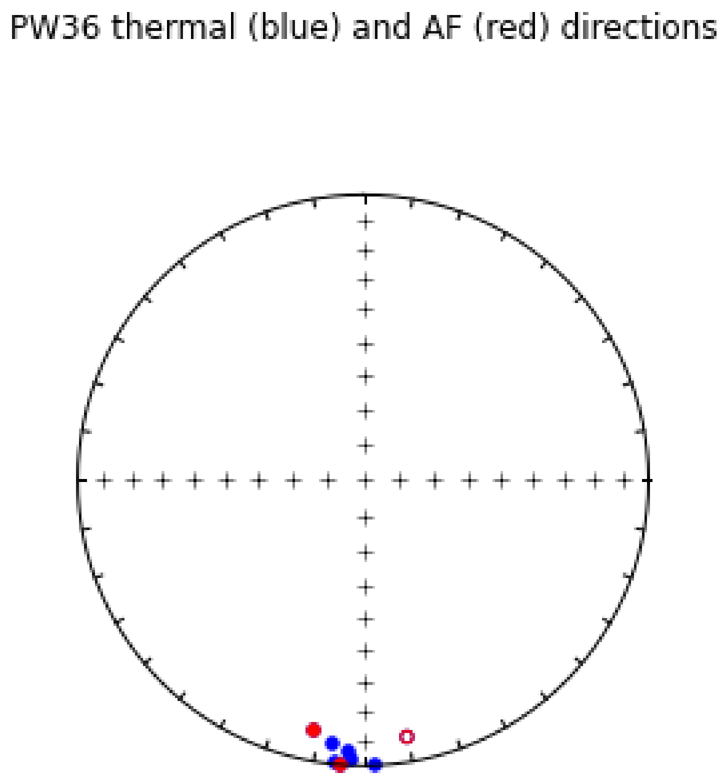


One outlier that is farther away from the rest is pruned from the data. Demag shows this sample beginning to arc towards the mean at high temperatures but goes unstable before it reaches.

AF demag

```
In [122]: PW36_AFdata=pd.read_csv('../Data/Botswana_AF/PW36/pmag_specimens.txt',
                                  sep='\t',header=1)
          PW36_AFtc = PW36_AFdata[PW36_AFdata['specimen_tilt_correction'] == 100]
          PW36_AFtc.reset_index(drop=True, inplace=True)

          PW36_AFtc_dir=[]
          for n in range(len(PW36_AFtc)):
```

```
    Dec,Inc=PW36_AFtc['specimen_dec'][n],PW36_AFtc['specimen_inc'][n]
    PW36_AFtc_dir.append([Dec,Inc,1.])
PW36_AFtc_mean=pmag.fisher_mean(PW36_AFtc_dir)

#Drop redundant specimens from same sample out of AF group.
PW36_AFtc_edit = PW36_AFtc
PW36_AFtc_edit = PW36_AFtc_edit.drop(0)
PW36_AFtc_edit = PW36_AFtc_edit.drop(3)
PW36_AFtc_edit = PW36_AFtc_edit.drop(4)
PW36_AFtc_edit = PW36_AFtc_edit.drop(5)
PW36_AFtc_edit = PW36_AFtc_edit.drop(6)
PW36_AFtc_edit.reset_index(drop=True, inplace=True)
PW36_AFtc_edit_dir=[]
for n in range(len(PW36_AFtc_edit)):
    Dec,Inc=PW36_AFtc_edit['specimen_dec'][n],PW36_AFtc_edit['specimen_inc'][n]
    PW36_AFtc_edit_dir.append([Dec,Inc,1.])
PW36_AFtc_edit_mean=pmag.fisher_mean(PW36_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW36_AFtc_dir,color='blue')
IPmag.iplotDI(PW36_AFtc_edit_dir,color='red')
plt.title('PW36 thermal (blue) and AF (red) directions')
plt.show()
```

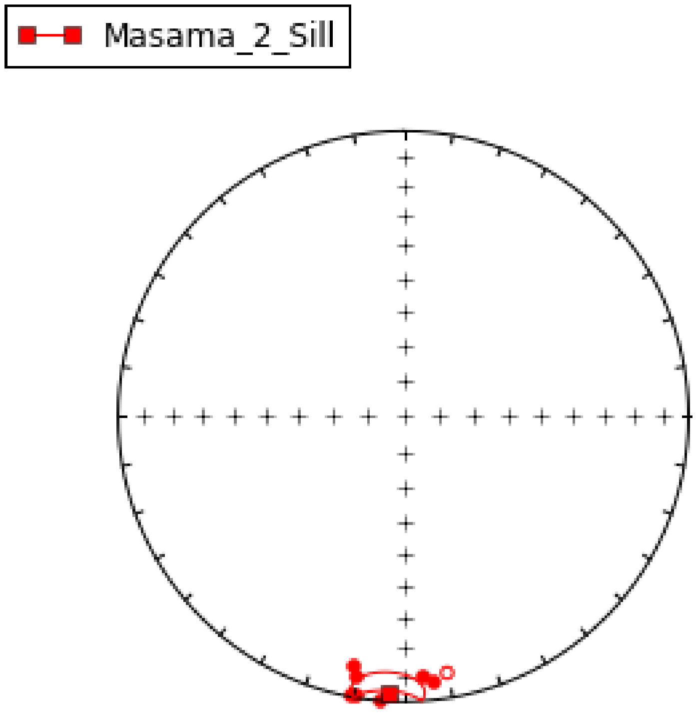


Three AF specimens (red) that do not are not redundant with the included thermal specimens are included in the intrusion mean direction.

Masama 2 Sill combined mean

```
In [123]: Masama_2_Sill=[]
      Masama_2_Sill = PW36_tc_edit_dir + PW36_AFtc_edit_dir
      Masama_2_Sill_mean=pmag.fisher_mean(Masama_2_Sill)

      fignum = 1
      plt.figure(num=fignum,figsize=(5,5))
      IPmag.iplotNET(1)
      IPmag.iplotDI(Masama_2_Sill,color='red')
      IPmag.iplotDImean(Masama_2_Sill_mean['dec'],Masama_2_Sill_mean['inc'],
                        Masama_2_Sill_mean["alpha95"],color='r',marker='s',
```

```
                    label='Masama_2_Sill')
```




```
In [124]: Intrusion_mean_directions.loc['Masama_2_Sill'] = pd.Series({
              'Intrusion_name':'Masama_2_Sill','sites_used':'PW36_ALL',
              'site_lat':Site_Locations['LAT(WGS84)'][35],
              'site_long':Site_Locations['LONG(WGS84)'][35],
              'n':int(Masama_2_Sill_mean['n']),
              'dec_tc':round(Masama_2_Sill_mean['dec'],1),
              'inc_tc':round(Masama_2_Sill_mean['inc'],1),
              'a_95':round(Masama_2_Sill_mean['alpha95'],1),
              'k':round(Masama_2_Sill_mean['k'],1),
              'dip_direction':10.4,'dip':8})
      Intrusion_mean_directions.ix['Masama_2_Sill']

Out[124]: Intrusion_name    Masama_2_Sill
      sites_used              PW36_ALL
```

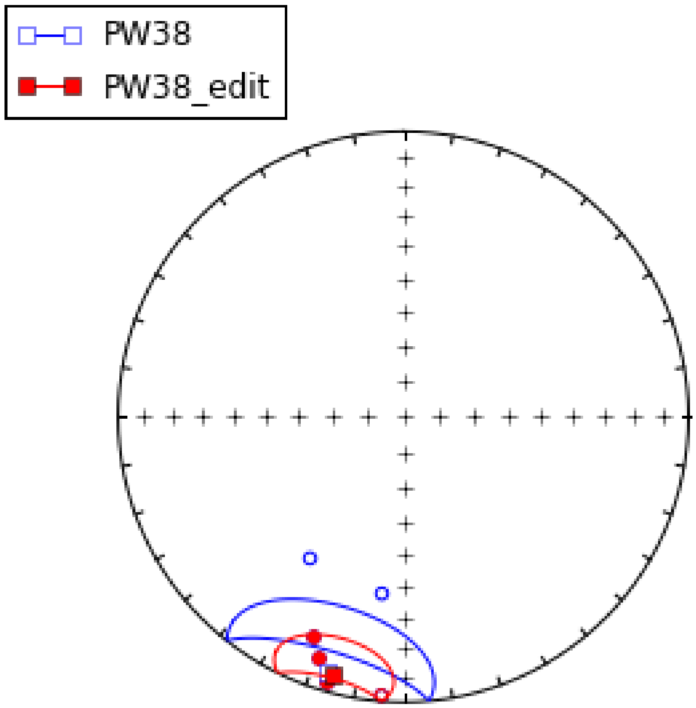
```
site_lat              -23.81453
site_long              26.73503
n                             8
dec_tc                    183.2
inc_tc                      3.5
a_95                        7.7
k                          53.1
date                        NaN
date_error                  NaN
dip_direction              10.4
dip                           8
Name: Masama_2_Sill, dtype: object
```

3.4.31 Dibete Kop Sill - PW38

Thermal demag

```
In [125]: PW38_tc_dir=[]
      for n in range(len(PW38_tc)):
          Dec,Inc=PW38_tc['specimen_dec'][n],PW38_tc['specimen_inc'][n]
          PW38_tc_dir.append([Dec,Inc,1.])
      PW38_tc_mean=pmag.fisher_mean(PW38_tc_dir)

      PW38_tc_edit = PW38_tc
      PW38_tc_edit = PW38_tc_edit.drop(0)
      PW38_tc_edit = PW38_tc_edit.drop(1)
      PW38_tc_edit.reset_index(inplace=True)

      PW38_tc_edit_dir=[]
      for n in range(len(PW38_tc_edit)):
          Dec,Inc=PW38_tc_edit['specimen_dec'][n],PW38_tc_edit['specimen_inc'][n]
          PW38_tc_edit_dir.append([Dec,Inc,1.])
      PW38_tc_edit_mean=pmag.fisher_mean(PW38_tc_edit_dir)

      fignum = 1
      plt.figure(num=fignum,figsize=(5,5))
      IPmag.iplotNET(1)
      IPmag.iplotDI(PW38_tc_dir,color='b')
      IPmag.iplotDImean(PW38_tc_mean['dec'],PW38_tc_mean['inc'],
                        PW38_tc_mean["alpha95"],color='b',marker='s',
                        label='PW38')
      IPmag.iplotDI(PW38_tc_edit_dir,color='r')
      IPmag.iplotDImean(PW38_tc_edit_mean['dec'],PW38_tc_edit_mean['inc'],
                        PW38_tc_edit_mean["alpha95"],color='r',marker='s',
```

```
                         label='PW38_edit')
```

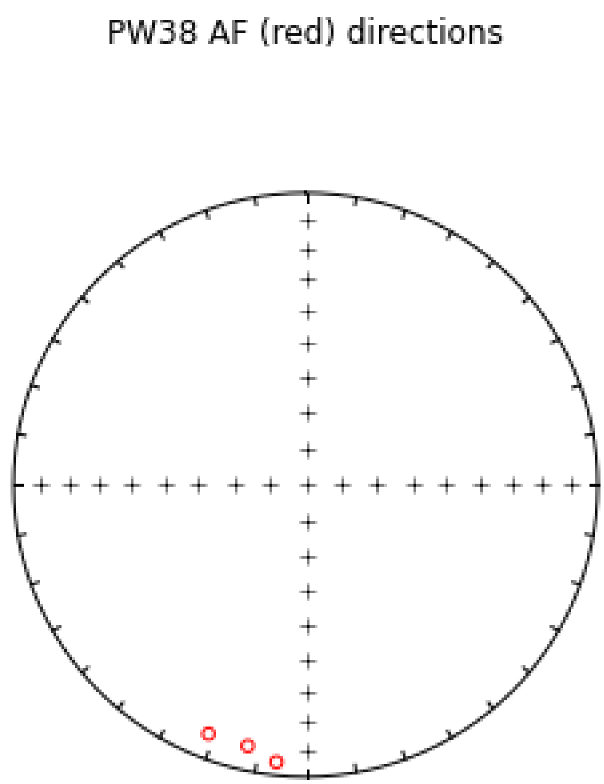


Two samples with steeper inclinations are considered outliers and are excluded - they had slightly higher intensities, perhaps due to lightning.

AF demag

```
In [126]: PW38_AFdata=pd.read_csv('../Data/Botswana_AF/PW38/pmag_specimens.txt',
                                sep='\t',header=1)
      PW38_AFtc = PW38_AFdata[PW38_AFdata['specimen_tilt_correction'] == 100]
      PW38_AFtc.reset_index(drop=True, inplace=True)

      PW38_AFtc_dir=[]
      for n in range(len(PW38_AFtc)):
          Dec,Inc=PW38_AFtc['specimen_dec'][n],PW38_AFtc['specimen_inc'][n]
```

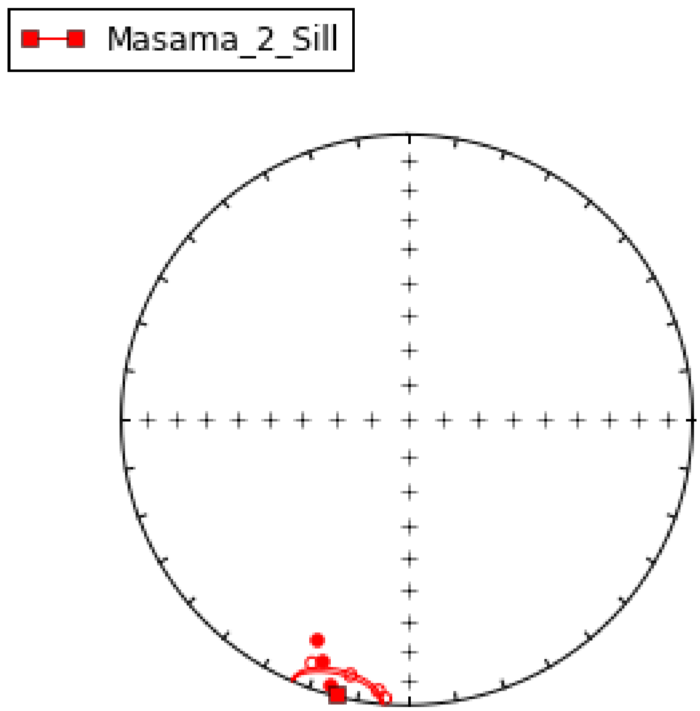
```
    PW38_AFtc_dir.append([Dec,Inc,1.])
PW38_AFtc_mean=pmag.fisher_mean(PW38_AFtc_dir)
#drop redundant specimens from the sam samples
PW38_AFtc_edit = PW38_AFtc
PW38_AFtc_edit = PW38_AFtc_edit.drop(2)
PW38_AFtc_edit = PW38_AFtc_edit.drop(3)
PW38_AFtc_edit = PW38_AFtc_edit.drop(4)
PW38_AFtc_edit = PW38_AFtc_edit.drop(5)
#drop outlying specimen
PW38_AFtc_edit = PW38_AFtc_edit.drop(7)
PW38_AFtc_edit.reset_index(inplace=True)
PW38_AFtc_edit_dir=[]
for n in range(len(PW38_AFtc_edit)):
    Dec,Inc=PW38_AFtc_edit['specimen_dec'][n],PW38_AFtc_edit['specimen_inc'][n]
    PW38_AFtc_edit_dir.append([Dec,Inc,1.])
PW38_AFtc_edit_mean=pmag.fisher_mean(PW38_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW38_AFtc_edit_dir,color='red')
plt.title('PW38 AF (red) directions')
plt.show()
```



Data from unique AF samples are combined with the thermal samples for the mean.

Dibete Kop Sill - PW38_ALL

```
In [127]: Dibete_Kop_Sill=[]
          Dibete_Kop_Sill = PW38_tc_edit_dir + PW38_AFtc_edit_dir
          Dibete_Kop_Sill_mean=pmag.fisher_mean(Dibete_Kop_Sill)

          fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
          IPmag.iplotNET(1)
          IPmag.iplotDI(Dibete_Kop_Sill,color='red')
          IPmag.iplotDImean(Dibete_Kop_Sill_mean['dec'],Dibete_Kop_Sill_mean['inc'],
                            Dibete_Kop_Sill_mean["alpha95"],color='r',marker='s',
                            label='Masama_2_Sill')
```



This mean is added to the compilation of Umkondo results.

```
In [128]: Intrusion_mean_directions.loc['Dibete_Kop_Sill'] = pd.Series({
              'Intrusion_name':'Dibete_Kop_Sill','sites_used':'PW38_ALL',
              'site_lat':Site_Locations['LAT(WGS84)'][37],
              'site_long':Site_Locations['LONG(WGS84)'][37],
              'n':int(Dibete_Kop_Sill_mean['n']),
              'dec_tc':round(Dibete_Kop_Sill_mean['dec'],1),
              'inc_tc':round(Dibete_Kop_Sill_mean['inc'],1),
              'a_95':round(Dibete_Kop_Sill_mean['alpha95'],1),
              'k':round(Dibete_Kop_Sill_mean['k'],1),
              'dip_direction':0,'dip':0})
      Intrusion_mean_directions.ix['Dibete_Kop_Sill']
```

```
Out[128]: Intrusion_name     Dibete_Kop_Sill
      sites_used            PW38_ALL
```

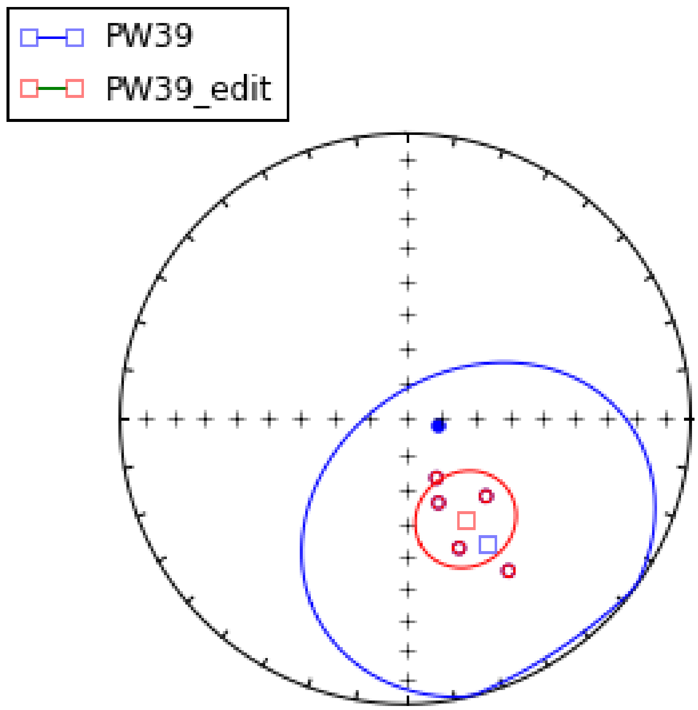
```
site_lat                -23.78171
site_long                26.56308
n                               7
dec_tc                      194.4
inc_tc                        0.8
a_95                          9.4
k                            42.4
date                          NaN
date_error                    NaN
dip_direction                   0
dip                             0
Name: Dibete_Kop_Sill, dtype: object
```

3.4.32 Marseilles Hill Sill - PW39

Thermal demag Consistent bahavior but appears to be close to the Paleoproterozoic direction described in Gose et al. (2006).

```
In [129]: PW39_tc_dir=[]
          for n in range(len(PW39_tc)):
              Dec,Inc=PW39_tc['specimen_dec'][n],PW39_tc['specimen_inc'][n]
              PW39_tc_dir.append([Dec,Inc,1.])
          PW39_tc_mean=pmag.fisher_mean(PW39_tc_dir)

          PW39_tc_edit = PW39_tc
          PW39_tc_edit = PW39_tc_edit.drop(4)
          PW39_tc_edit.reset_index(inplace=True)

          PW39_tc_edit_dir=[]
          for n in range(len(PW39_tc_edit)):
              Dec,Inc=PW39_tc_edit['specimen_dec'][n],PW39_tc_edit['specimen_inc'][n]
              PW39_tc_edit_dir.append([Dec,Inc,1.])
          PW39_tc_edit_mean=pmag.fisher_mean(PW39_tc_edit_dir)

          fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
          IPmag.iplotNET(1)
          IPmag.iplotDI(PW39_tc_dir,color='b')
          IPmag.iplotDImean(PW39_tc_mean['dec'],PW39_tc_mean['inc'],
                            PW39_tc_mean["alpha95"],color='b',marker='s',
                            label='PW39')
          IPmag.iplotDI(PW39_tc_edit_dir,color='r')
          IPmag.iplotDImean(PW39_tc_edit_mean['dec'],PW39_tc_edit_mean['inc'],
                            PW39_tc_edit_mean["alpha95"],color='r',marker='s',
```

```
                              label='PW39_edit')
```

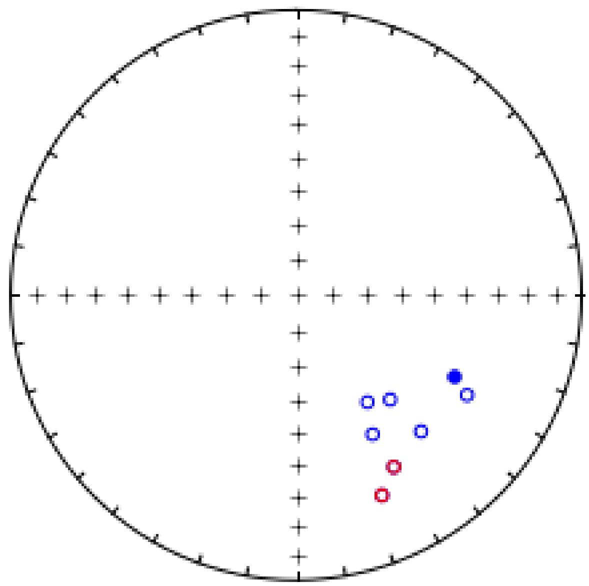


There is good agreement between samples with the exception of one point which is considered an outlier.

AF demag

```
In [130]: PW39_AFdata=pd.read_csv('../Data/Botswana_AF/PW39/pmag_specimens.txt',
                              sep='\t',header=1)
      PW39_AFtc = PW39_AFdata[PW39_AFdata['specimen_tilt_correction'] == 100]
      PW39_AFtc.reset_index(drop=True, inplace=True)

      PW39_AFtc_dir=[]
      for n in range(len(PW39_AFtc)):
          Dec,Inc=PW39_AFtc['specimen_dec'][n],PW39_AFtc['specimen_inc'][n]
```

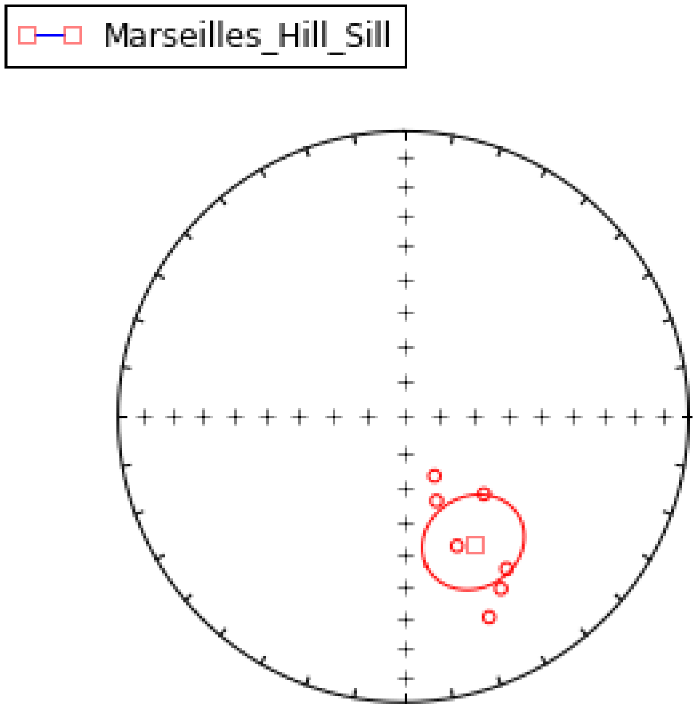
```
    PW39_AFtc_dir.append([Dec,Inc,1.])
PW39_AFtc_mean=pmag.fisher_mean(PW39_AFtc_dir)

#Drop redundant specimens from same sample out of AF group.
PW39_AFtc_edit = PW39_AFtc
PW39_AFtc_edit = PW39_AFtc_edit.drop(2)
PW39_AFtc_edit = PW39_AFtc_edit.drop(3)
PW39_AFtc_edit = PW39_AFtc_edit.drop(4)
PW39_AFtc_edit = PW39_AFtc_edit.drop(5)
PW39_AFtc_edit = PW39_AFtc_edit.drop(7)
#drop an additional sample far from the mean grouping
PW39_AFtc_edit = PW39_AFtc_edit.drop(6)
PW39_AFtc_edit.reset_index(drop=True, inplace=True)
PW39_AFtc_edit_dir=[]
for n in range(len(PW39_AFtc_edit)):
    Dec,Inc=PW39_AFtc_edit['specimen_dec'][n],PW39_AFtc_edit['specimen_inc'][n]
    PW39_AFtc_edit_dir.append([Dec,Inc,1.])
PW39_AFtc_edit_mean=pmag.fisher_mean(PW39_AFtc_edit_dir)

fignum = 1
plt.figure(num=fignum,figsize=(5,5))
IPmag.iplotNET(1)
IPmag.iplotDI(PW39_AFtc_dir,color='blue')
IPmag.iplotDI(PW39_AFtc_edit_dir,color='red')
plt.show()
```



Marseilles Hill Sill - PW39_ALL There is good agreement between AF and thermal demagnetization data. The mean direction is far from Umkondo, leading us to believe that this sill is not part of the Umkondo LIP.

```
In [131]: Marseilles_Hill_Sill=[]
          Marseilles_Hill_Sill = PW39_tc_edit_dir + PW39_AFtc_edit_dir
          Marseilles_Hill_Sill_mean=pmag.fisher_mean(Marseilles_Hill_Sill)

          fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
          IPmag.iplotNET(1)
          IPmag.iplotDI(Marseilles_Hill_Sill,color='red')
          IPmag.iplotDImean(Marseilles_Hill_Sill_mean['dec'],
                            Marseilles_Hill_Sill_mean['inc'],
```

```
                Marseilles_Hill_Sill_mean["alpha95"],color='r',
                marker='s',label='Marseilles_Hill_Sill')
```



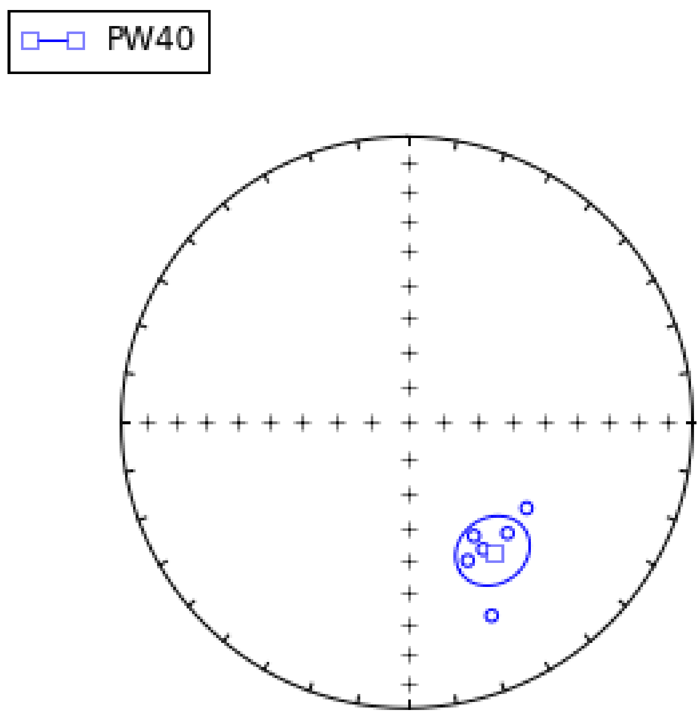
Not added to the Umkondo compilation because this intrusion is likely older in age (added to unknown and older intrusion table).

```
In [132]: unknown_intrusions.loc['Marseilles_Hill_Sill'] = pd.Series({'site_ID':
                    'PW39_ALL',
                    'site_lat':Site_Locations['LAT(WGS84)'][38],
                    'site_long':Site_Locations['LAT(WGS84)'][38],
                    'n':Marseilles_Hill_Sill_mean['n'],
                    'dec_tc':round(Marseilles_Hill_Sill_mean['dec'],1),
                    'inc_tc':round(Marseilles_Hill_Sill_mean['inc'],1),
                    'a_95':round(Marseilles_Hill_Sill_mean['alpha95'],1),
                    'k':round(Marseilles_Hill_Sill_mean['k'],1)})
```

3.4.33 Sepitswane Sill - PW40

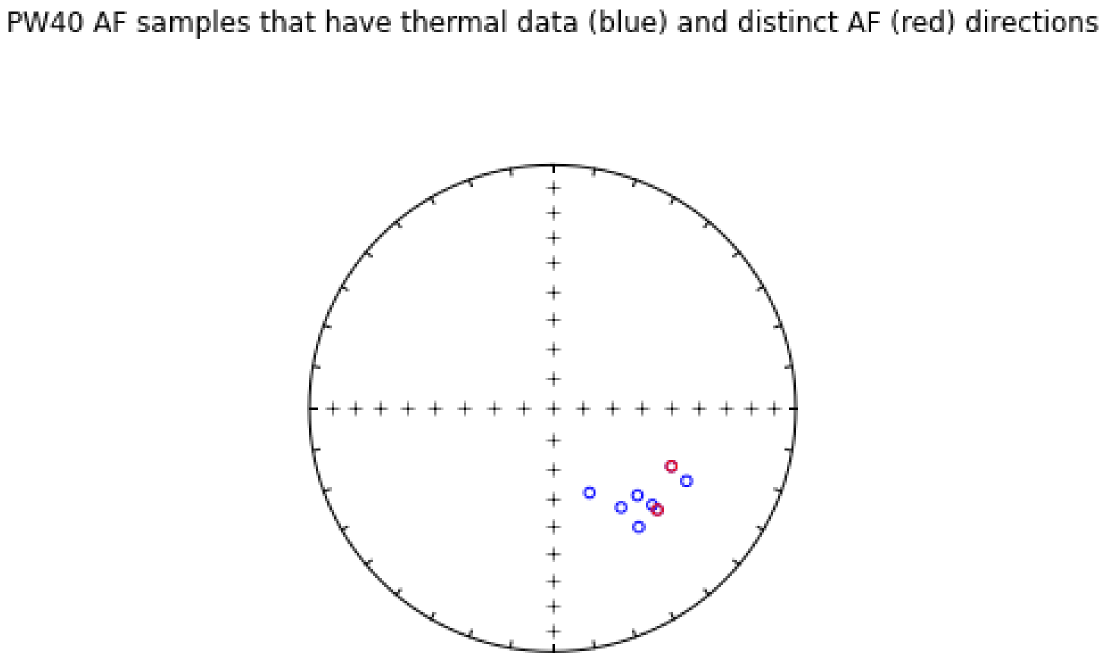
Thermal demag Great sample behavior and consistency. However, based on the direction this is likely not an Umkondo sill, but probably an older sill as described in Gose et al. (2006).

```
In [133]: PW40_tc_dir=[]
          for n in range(len(PW38_tc)):
              Dec,Inc=PW40_tc['specimen_dec'][n],PW40_tc['specimen_inc'][n]
              PW40_tc_dir.append([Dec,Inc,1.])
          PW40_tc_mean=pmag.fisher_mean(PW40_tc_dir)

          fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
          IPmag.iplotNET(1)
          IPmag.iplotDI(PW40_tc_dir,color='b')
          IPmag.iplotDImean(PW40_tc_mean['dec'],PW40_tc_mean['inc'],
                            PW40_tc_mean["alpha95"],color='b',marker='s',
                            label='PW40')
```



AF demag

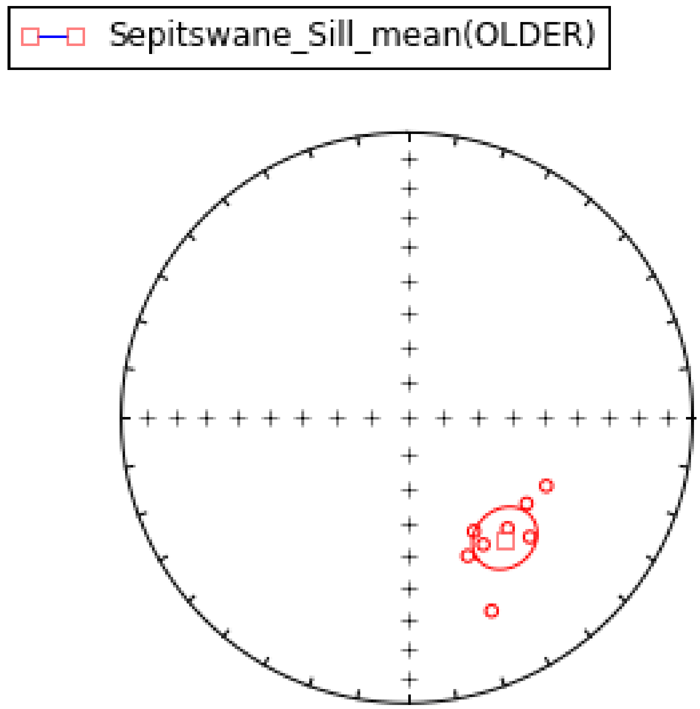
```
In [134]: PW40_AFdata=pd.read_csv('../Data/Botswana_AF/PW40/pmag_specimens.txt',
                                  sep='\t',header=1)
          PW40_AFtc = PW40_AFdata[PW40_AFdata['specimen_tilt_correction'] == 100]
          PW40_AFtc.reset_index(drop=True, inplace=True)

          PW40_AFtc_dir=[]
          for n in range(len(PW40_AFtc)):
              Dec,Inc=PW40_AFtc['specimen_dec'][n],PW40_AFtc['specimen_inc'][n]
              PW40_AFtc_dir.append([Dec,Inc,1.])
          PW40_AFtc_mean=pmag.fisher_mean(PW40_AFtc_dir)

          #Drop redundant specimens from same sample out of AF group.
          PW40_AFtc_edit = PW40_AFtc
          PW40_AFtc_edit = PW40_AFtc_edit.drop(0)
          PW40_AFtc_edit = PW40_AFtc_edit.drop(1)
          PW40_AFtc_edit = PW40_AFtc_edit.drop(4)
          PW40_AFtc_edit = PW40_AFtc_edit.drop(5)
          PW40_AFtc_edit = PW40_AFtc_edit.drop(6)
          PW40_AFtc_edit = PW40_AFtc_edit.drop(7)
          PW40_AFtc_edit.reset_index(drop=True, inplace=True)
          PW40_AFtc_edit_dir=[]
          for n in range(len(PW40_AFtc_edit)):
              Dec,Inc=PW40_AFtc_edit['specimen_dec'][n],PW40_AFtc_edit['specimen_inc'][n]
              PW40_AFtc_edit_dir.append([Dec,Inc,1.])
          PW40_AFtc_edit_mean=pmag.fisher_mean(PW40_AFtc_edit_dir)

          fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
          IPmag.iplotNET(1)
          IPmag.iplotDI(PW40_AFtc_dir,color='blue')
          IPmag.iplotDI(PW40_AFtc_edit_dir,color='red')
          plt.title('PW40 AF samples that have thermal data (blue) and distinct AF (red) direction
          plt.show()
```



The two specimens that are not redundant are well within the thermal mean grouping. It is very likely that this (like PW39) is an older intrusion unrelated to the Umkondo LIP.

Sepitswane Sill mean - PW40

```
In [135]: Sepitswane_Sill=[]
          Sepitswane_Sill = PW40_tc_dir + PW40_AFtc_edit_dir
          Sepitswane_Sill_mean=pmag.fisher_mean(Sepitswane_Sill)

          fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
          IPmag.iplotNET(1)
          IPmag.iplotDI(Sepitswane_Sill,color='red')
          IPmag.iplotDImean(Sepitswane_Sill_mean['dec'],
                            Sepitswane_Sill_mean['inc'],
                            Sepitswane_Sill_mean["alpha95"],color='r',
                            marker='s',label='Sepitswane_Sill_mean(OLDER)')
```



Not added to table because of its similarity to older paleomagnetic directions (but added to the unknown and older intrusion table).

```
In [136]: unknown_intrusions.loc['Sepitswane_Sill'] = pd.Series({'site_ID':
                              'PW40_ALL',
                              'site_lat':Site_Locations['LAT(WGS84)'][39],
                              'site_long':Site_Locations['LAT(WGS84)'][39],
                              'n':Sepitswane_Sill_mean['n'],
                              'dec_tc':round(Sepitswane_Sill_mean['dec'],1),
                              'inc_tc':round(Sepitswane_Sill_mean['inc'],1),
                              'a_95':round(Sepitswane_Sill_mean['alpha95'],1),
                              'k':round(Sepitswane_Sill_mean['k'],1)})
```

3.5 Botswana Umkondo summary (Thermal/AF demagnetization results combined with existing data from the same intrusions)

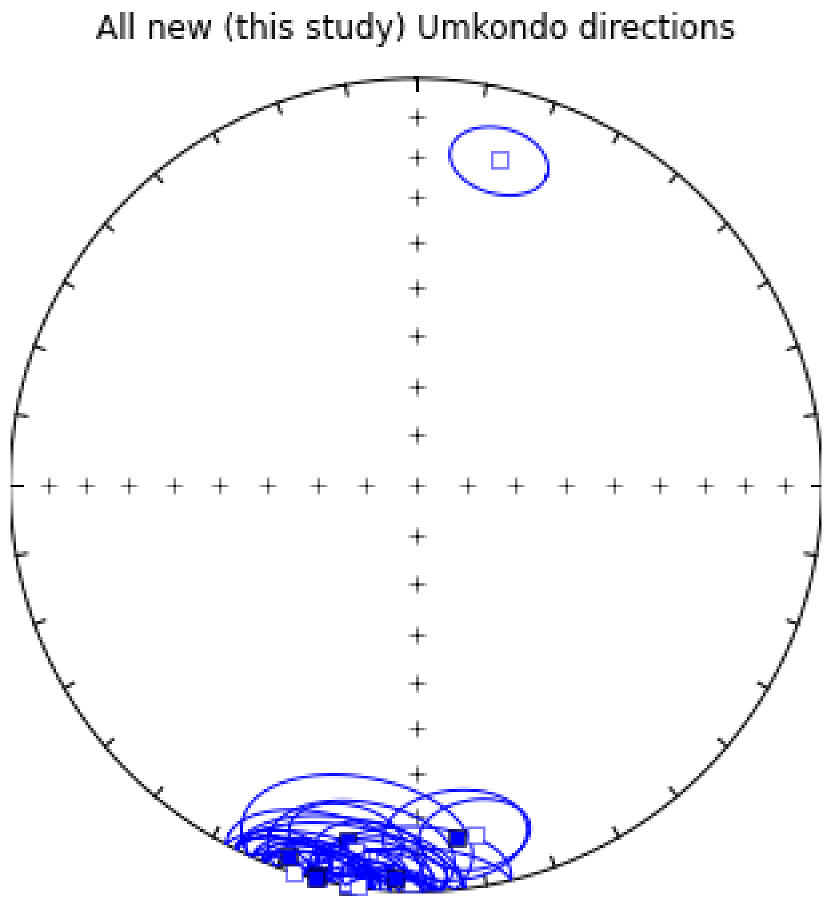
Plot up all of the results together and then build another dataframe with data in pole space.

```
In [137]: # Calculate/Add VGP information
          Intrusions_newBot_VGPs_rnd=Intrusion_mean_directions
          IPmag.VGP_calc(Intrusions_newBot_VGPs_rnd)
          # Changing two labels to be the same as in the "Published Data Compilation"
          Intrusions_newBot_VGPs_rnd=Intrusions_newBot_VGPs_rnd.rename(columns=
                                                                      {'sites_used':
                                                                       'site_ID'})
          Intrusions_newBot_VGPs_rnd=Intrusions_newBot_VGPs_rnd.drop('Intrusion_name',1)
          Intrusions_newBot_VGPs_rnd
```

Out[137]:

	site_ID	site_lat	site_long	n	dec_tc	inc_tc	a_95	k	date	date_error	di
Kgale_Peak_Sill	PW1_ALL and PW2_AF	-24.68781	25.86215	12	189.9	0.4	6.3	48.8	1108.0	0.9	
Rasemong_Sill	PW5_ALL	-24.72715	25.77590	8	14.4	-18.6	8.1	48.0	NaN	NaN	
Metsemotlhaba_River_Sill	PW6_ALL	-24.54694	25.80892	7	180.6	-2.7	14.4	18.5	NaN	NaN	
Mabogoapitse_Hill_Sill	PW7_ALL and PW8_AF	-24.47402	25.59705	9	184.6	5.1	9.1	33.2	NaN	NaN	
Semarule_Hill_Sill	PW9_ALL	-24.45268	25.57415	5	186.8	3.8	9.1	72.2	NaN	NaN	
Rapitsane_Sill	PW10_ALL	-24.41968	25.58463	8	197.7	-0.2	8.5	43.3	NaN	NaN	
Suping_Sill	PW11_ALL and JP15	-24.32765	25.53224	16	187.2	-9.2	8.4	20.2	NaN	NaN	
Mogatelwane 2 Sill	PW15_ALL	-24.18042	25.69191	6	193.5	-2.8	14.7	21.6	NaN	NaN	
Mosolotsane_1_Sill	PW21_AF, PW22_ALL, and JP(22,23,24)	-22.90699	26.38929	27	186.1	-5.6	4.6	36.9	1109.3	0.6	
Mosolotsane_5_Sill	PW23_ALL	-22.90330	26.37027	7	188.5	-7.9	14.2	19.1	NaN	NaN	
Mosolotsane_4_Sill	PW24_AF	-22.89467	26.37410	8	185.2	-2.5	7.9	50.3	NaN	NaN	
Mosolotsane_6_Sill	PW25_AF	-22.89550	26.36726	5	191.2	11.8	9.0	72.7	NaN	NaN	
Mosolotsane_3_Sill	PW26_AF	-22.89259	26.38113	4	189.8	-0.9	16.7	31.3	NaN	NaN	
Mosolotsane_2_Sill	PW27_AF	-22.89228	26.38196	8	187.6	2.0	5.6	97.5	NaN	NaN	
Shoshong_Sill	PW28_AF and JP(26,31,33,34)	-23.00519	26.48383	33	191.5	-5.4	3.1	65.2	1109.3	0.4	
Phage_Sill	PW29_AF	-22.77939	26.39372	8	194.0	-0.8	7.8	50.9	NaN	NaN	
Moijabana_Sill	PW30_AF	-22.64213	26.44260	5	189.4	-10.0	17.9	19.2	NaN	NaN	
Mokgware_Sill	PW31_ALL and JP30	-22.70685	26.61142	13	199.0	3.8	6.5	42.2	1112.0	0.5	
Sepatamorire_Sill	PW32_ALL	-22.33543	26.82285	8	194.1	1.5	8.3	45.6	NaN	NaN	
Palapye_dike	PW33_AF	-22.57771	27.28736	7	173.6	13.9	11.4	29.0	NaN	NaN	
Masama_1_Sill	PW34_ALL	-23.81626	26.73769	8	170.5	-13.6	8.8	40.3	NaN	NaN	
Masama_3_Sill	PW35_AF and PW37_ALL	-23.81403	26.73541	13	188.5	-0.7	4.8	75.4	NaN	NaN	
Masama_2_Sill	PW36_ALL	-23.81453	26.73503	8	183.2	3.5	7.7	53.1	NaN	NaN	
Dibete_Kop_Sill	PW38_ALL	-23.78171	26.56308	7	194.4	0.8	9.4	42.4	NaN	NaN	

```
In [138]: fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
          IPmag.iplotNET(1)
          for n in range(len(Intrusion_mean_directions)):
              IPmag.iplotDImean(Intrusion_mean_directions['dec_tc'][n],
                                Intrusion_mean_directions['inc_tc'][n],
                                Intrusion_mean_directions['a_95'][n],
                                color='b',marker='s',label='')
          plt.title('All new (this study) Umkondo directions')
          plt.show()
```



3.6 Low Temperature Magnetizations from New Umkondo Sites

A summary of the low-temperature components found in specimens that had more than one component. There is no real consistency between sites, but some sites have an internally consistent low temperature overprint. Our interpretation is that the low temperature components from the intrusions in Botswana can be attributed to viscous remanent magnetizations, with no paleomagnetic significance.

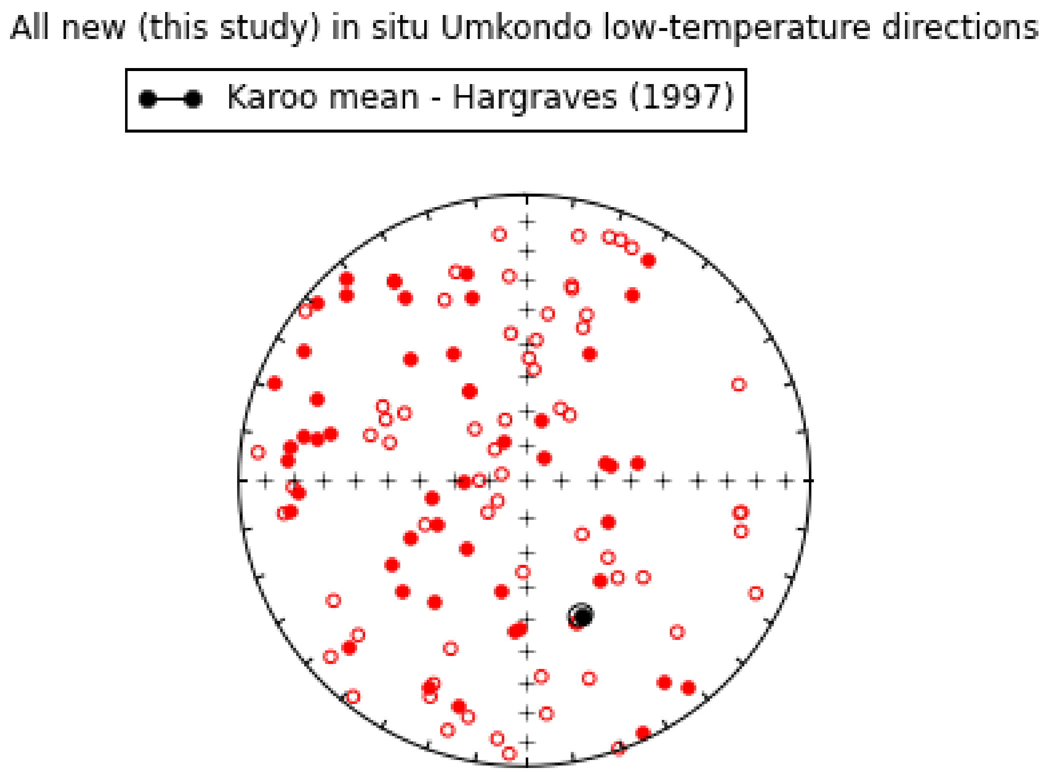
```
In [139]: #The sites that are *not* loaded into this dataset contained only...
          #...single-component remanences discussed earlier in this notebook
          #PW1L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW1/pmag_specimens.txt
          PW3L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW3/pmag_specimens.txt'
          PW4L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW4/pmag_specimens.txt'
```

```
PW5L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW5/pmag_specimens.txt'
PW6L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW6/pmag_specimens.txt'
PW7L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW7/pmag_specimens.txt'
PW9L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW9/pmag_specimens.txt'
PW10L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW10/pmag_specimens.tx
PW11L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW11/pmag_specimens.tx
PW13L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW13/pmag_specimens.tx
PW15L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW15/pmag_specimens.tx
PW18L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW18/pmag_specimens.tx
PW19L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW19/pmag_specimens.tx
PW20L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW20/pmag_specimens.tx
#PW22L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW22/pmag_specimens.t
PW23L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW23/pmag_specimens.tx
PW24L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW24/pmag_specimens.tx
PW25L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW25/pmag_specimens.tx
PW26L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW26/pmag_specimens.tx
PW27L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW27/pmag_specimens.tx
PW28L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW28/pmag_specimens.tx
PW29L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW29/pmag_specimens.tx
#PW30L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW30/pmag_specimens.t
PW31L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW31/pmag_specimens.tx
PW32L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW32/pmag_specimens.tx
PW34L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW34/pmag_specimens.tx
PW36L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW36/pmag_specimens.tx
PW37L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW37/pmag_specimens.tx
PW38L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW38/pmag_specimens.tx
PW39L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW39/pmag_specimens.tx
PW40L_data=pd.read_csv('../Data/Botswana_thermal/Botswana_data_LT/PW40/pmag_specimens.tx

PW_ALL_LT=pd.tools.merge.concat([PW3L_data,PW4L_data,PW5L_data,PW6L_data,
                                 PW7L_data,PW9L_data,PW10L_data,PW11L_data,
                                 PW13L_data,PW15L_data,PW18L_data,PW19L_data,
                                 PW20L_data,PW23L_data,PW24L_data,PW25L_data,
                                 PW26L_data,PW27L_data,PW28L_data,PW29L_data,
                                 PW31L_data,PW32L_data,PW34L_data,PW36L_data,
                                 PW37L_data,PW38L_data,PW39L_data,PW40L_data],
                                axis=0)
PW_ALL_LT_tc= PW_ALL_LT[PW_ALL_LT['specimen_tilt_correction'] == 100]
PW_ALL_LT_tc.reset_index(inplace=True, drop=True)
PW_ALL_LT_geo=PW_ALL_LT[PW_ALL_LT['specimen_tilt_correction'] == 0]
PW_ALL_LT_geo.reset_index(inplace=True, drop=True)
```

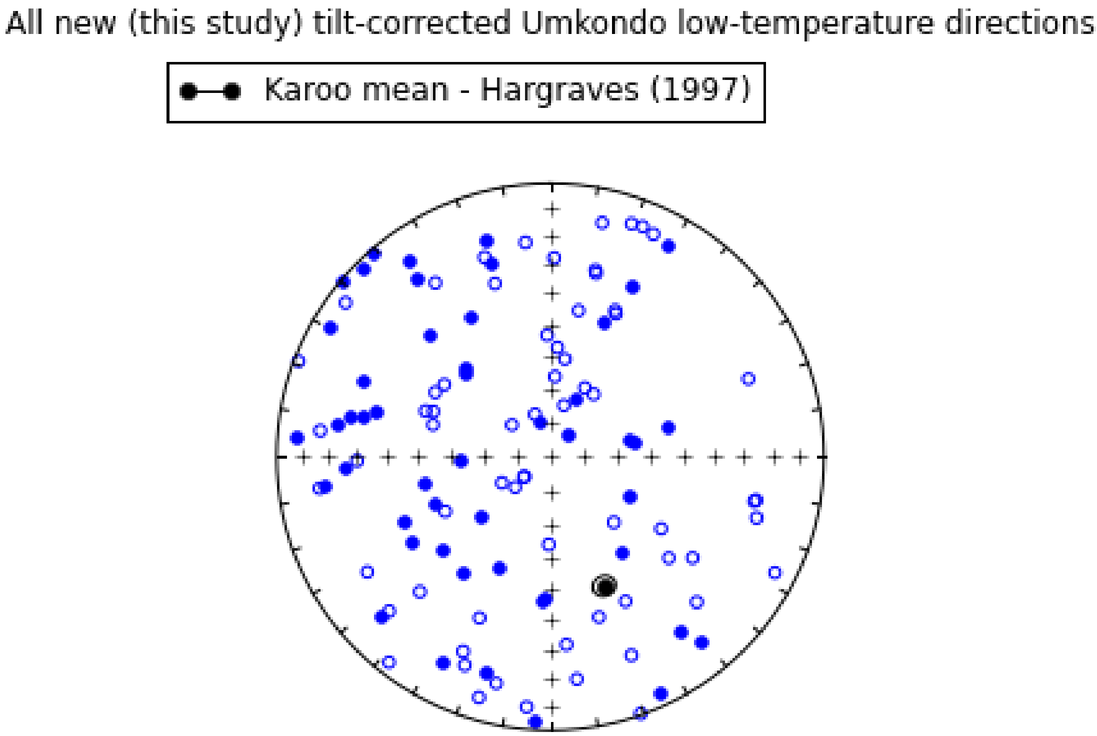
We will plot the Karoo LIP pole from Hargraves (1997) with the low temperature magnetizations from our new Botswana sites to see if their directions are similar.

Virtual geomagnetic poles have been derived from over 2000 samples from 691 sites in Karoo igneous rocks, ranging from basalts and dolerites through rhyolites, and at localities spread from Namibia through the central Karoo basin, Lesotho to the Lebombo. After appropriate analysis, poles derived from Lesotho lavas, Karoo dolerite dykes and sills, Sabie River basalts, Mashikiri and Letaba lavas, and Jozini rhyolites, when averaged, yielded a palaeomagnetic (south) pole at: N = 6, latitude = -69.2, longitude = 98.3, alpha = 3.3. (Hargraves, 1997)

```
In [140]: PW_ALL_LT_tc_dir = []
          PW_ALL_LT_geo_dir = []
          for n in range(len(PW_ALL_LT_tc)):
              Dec,Inc=PW_ALL_LT_tc['specimen_dec'][n],PW_ALL_LT_tc['specimen_inc'][n]
              PW_ALL_LT_tc_dir.append([Dec,Inc,1.])
          PW_ALL_LT_tc_mean=pmag.fisher_mean(PW_ALL_LT_tc_dir)
          for n in range(len(PW_ALL_LT_geo)):
              Dec,Inc=PW_ALL_LT_geo['specimen_dec'][n],PW_ALL_LT_geo['specimen_inc'][n]
              PW_ALL_LT_geo_dir.append([Dec,Inc,1.])
          PW_ALL_LT_geo_mean=pmag.fisher_mean(PW_ALL_LT_geo_dir)

          fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
          IPmag.iplotNET(1)
          IPmag.iplotDI(PW_ALL_LT_geo_dir,color='red')
          #Karoo mean from Hargraves (1997)
          IPmag.iplotDImean(157.4,47.6,3.2,color='k',label=
                            'Karoo mean - Hargraves (1997)')
          plt.title('All new (this study) in situ Umkondo low-temperature directions')
          plt.show()
```




```
In [141]: fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
          IPmag.iplotNET(1)
          IPmag.iplotDI(PW_ALL_LT_tc_dir,color='blue')
          #Karoo mean from Hargraves (1997)
          IPmag.iplotDImean(157.4,47.6,3.2,color='k',label=
                            'Karoo mean - Hargraves (1997)')
          #blue data points are tilt corrected...many sites had no tilting to correct
          plt.title('All new (this study) tilt-corrected Umkondo low-temperature directions')
          plt.show()
```



There is little in the way of consistency in the low-temperature overprints between the studied sites. Given the small tilt little difference between the geographic and tilt corrected results for the low-temperature overprints.

Maybe some of the overprints are associated with Karoo magmatism or older overprinting events, but there there is little consistency across the sites and there are few directions similar to the Karoo (˜180 Ma) magnetization mean. We note here that Seidel (2004) conducted a positive baked-contact test for a Karoo dike into the Timbavati gabbro (TG-01). This supports the fact that the Umkondo magnetization is older than ˜180 Ma. In general, the lack of consistency between the overprints suggests that their origin is due to lightning strikes or other processes such as wildfires both of which have significant impacts on the slowly evolving southern African landscape.

3.7 Unknown and Older Directions (from Gose et al. (2006) and this study)

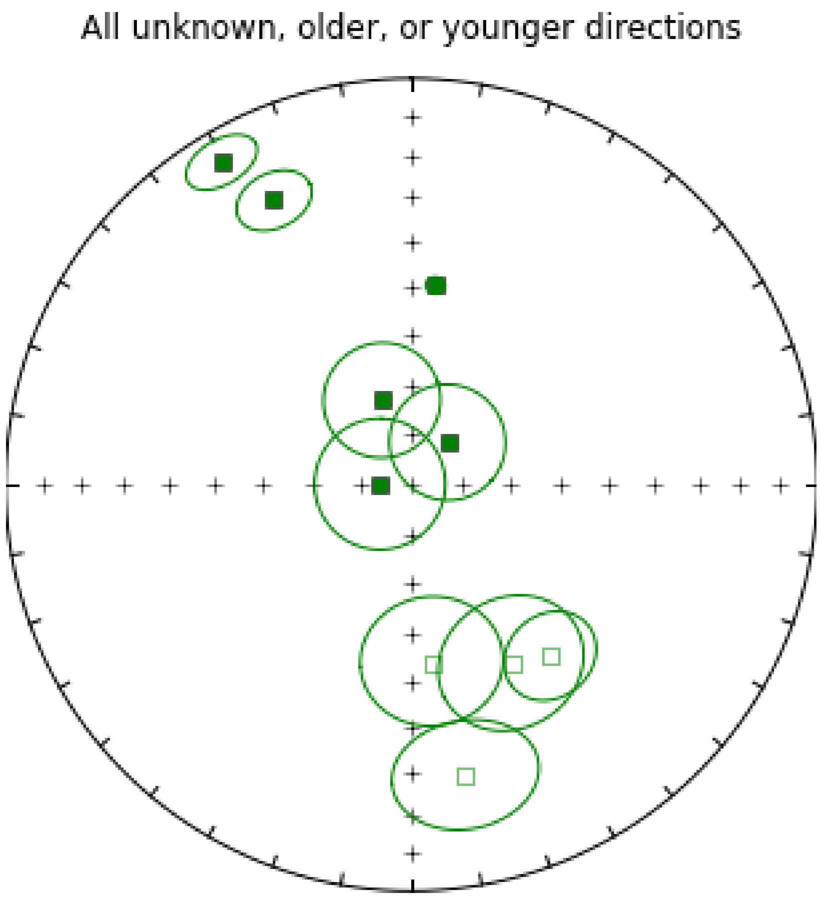
Below is a table of intrusions that are likely a different age than ˜1.1 Ga. This includes intrusions that are estimated to be ˜1.9 Ga (according to baddeleyite dates in Gose et al. (2006)): W03, JP9, WD28H, NB2, and JP11. If new paleomagnetic sites have similar directions to these sites it is likely that they belong to the older event.

```
In [142]: IPmag.VGP_calc(unknown_intrusions)
          unknown_intrusions
```

Out[142]:

	site_ID	site_lat	site_long	n	dec_tc	inc_tc	a_95	k	date	date_error	paleolatitude	pole_lat	pole_l
W03	W03	-25.70000	29.41000	6	271.2	83.70	13.1	27.1	NaN	NaN	77.548756	-24.796123	15.673
JP9	NaN	-24.96000	25.25000	5	334.2	23.18	6.6	134.4	NaN	NaN	12.084239	45.218799	348.079
JP11	NaN	-24.94000	25.30000	4	169.4	-27.76	12.7	53.1	NaN	NaN	-14.744434	-48.991236	189.569
J_M_A1	J_M_A1	-24.93000	25.30000	4	329.5	8.50	5.9	244.0	NaN	NaN	4.273513	48.396981	335.634
WD28H	WD28H	-24.50000	27.56000	11	340.8	72.00	11.6	16.6	NaN	NaN	56.982602	6.922676	17.160
L-2	L-2	-25.60000	29.62000	11	6.9	49.10	1.8	636.4	NaN	NaN	29.994221	34.013851	36.831
NB2	NB2	-22.56000	30.86000	8	173.4	-53.70	13.6	17.6	NaN	NaN	-34.241847	-32.852415	204.365
Mogatelwane_1_Sill	PW14_AF	-24.18075	25.68963	6	40.5	78.80	11.6	34.6	NaN	NaN	68.395906	-7.205706	39.636
Marseilles_Hill_Sill	PW39_ALL	-23.96123	-23.96123	7	150.8	-48.00	14.0	19.6	NaN	NaN	-29.043688	-30.014754	126.529
Sepitswane_Sill	PW40_ALL	-24.04815	-24.04815	8	140.9	-44.70	9.0	38.9	NaN	NaN	-26.325802	-27.030836	116.562

```
In [143]: IPmag.VGP_calc(unknown_intrusions)
          fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
          IPmag.iplotNET(1)
          for n in range(len(unknown_intrusions)):
              IPmag.iplotDImean(unknown_intrusions['dec_tc'][n],
                                unknown_intrusions['inc_tc'][n],
                                unknown_intrusions['a_95'][n],
                                color='g',marker='s',label='')
          plt.title('All unknown, older, or younger directions')
          plt.show()
```



Above are site mean directions from intrusions that are likely of younger and older age than the Umkondo event. The Karoo magnetization is similar to the steep SE and down mean from PW18, if this is indeed the case then PW18 could be an Umkondo sill overprinted by the Karoo event. Other magnetizations that are steep NW and down (along with the antipodal SE and up site means) could be Paleoproterozoic in age. One of the intrusions, WD28(H) is dated at 1871.9 ± 1.2 Ma.

4 Umkondo synthesis and grand mean pole

In [144]:
```
Old_Data_ALL = pickle.load(open('../Data/Pickle/cooling_unit_means_edit','rb'))
Intrusions_newBot_VGPs_rnd
```

Out[144]:

	site_ID	site_lat	site_long	n	dec_tc	inc_tc	a_95	k	date	date_error	di
Kgale_Peak_Sill	PW1_ALL and PW2_AF	-24.68781	25.86215	12	189.9	0.4	6.3	48.8	1108.0	0.9	
Rasemong_Sill	PW5_ALL	-24.72715	25.77590	8	14.4	-18.6	8.1	48.0	NaN	NaN	
Metsemotlhaba_River_Sill	PW6_ALL	-24.54694	25.80892	7	180.6	-2.7	14.4	18.5	NaN	NaN	
Mabogoapitse_Hill_Sill	PW7_ALL and PW8_AF	-24.47402	25.59705	9	184.6	5.1	9.1	33.2	NaN	NaN	
Semarule_Hill_Sill	PW9_ALL	-24.45268	25.57415	5	186.8	3.8	9.1	72.2	NaN	NaN	
Rapitsane_Sill	PW10_ALL	-24.41968	25.58463	8	197.7	-0.2	8.5	43.3	NaN	NaN	
Suping_Sill	PW11_ALL and JP15	-24.32765	25.53224	16	187.2	-9.2	8.4	20.2	NaN	NaN	
Mogatelwane 2 Sill	PW15_ALL	-24.18042	25.69191	6	193.5	-2.8	14.7	21.6	NaN	NaN	
Mosolotsane_1_Sill	PW21_AF, PW22_ALL, and JP(22,23,24)	-22.90699	26.38929	27	186.1	-5.6	4.6	36.9	1109.3	0.6	
Mosolotsane_5_Sill	PW23_ALL	-22.90330	26.37027	7	188.5	-7.9	14.2	19.1	NaN	NaN	
Mosolotsane_4_Sill	PW24_AF	-22.89467	26.37410	8	185.2	-2.5	7.9	50.3	NaN	NaN	
Mosolotsane_6_Sill	PW25_AF	-22.89550	26.36726	5	191.2	11.8	9.0	72.7	NaN	NaN	
Mosolotsane_3_Sill	PW26_AF	-22.89259	26.38113	4	189.8	-0.9	16.7	31.3	NaN	NaN	
Mosolotsane_2_Sill	PW27_AF	-22.89228	26.38196	8	187.6	2.0	5.6	97.5	NaN	NaN	
Shoshong_Sill	PW28_AF and JP(26,31,33,34)	-23.00519	26.48383	33	191.5	-5.4	3.1	65.2	1109.3	0.4	
Phage_Sill	PW29_AF	-22.77939	26.39372	8	194.0	-0.8	7.8	50.9	NaN	NaN	
Moijabana_Sill	PW30_AF	-22.64213	26.44260	5	189.4	-10.0	17.9	19.2	NaN	NaN	
Mokgware_Sill	PW31_ALL and JP30	-22.70685	26.61142	13	199.0	3.8	6.5	42.2	1112.0	0.5	
Sepatamorire_Sill	PW32_ALL	-22.33543	26.82285	8	194.1	1.5	8.3	45.6	NaN	NaN	
Palapye_dike	PW33_AF	-22.57771	27.28736	7	173.6	13.9	11.4	29.0	NaN	NaN	
Masama_1_Sill	PW34_ALL	-23.81626	26.73769	8	170.5	-13.6	8.8	40.3	NaN	NaN	
Masama_3_Sill	PW35_AF and PW37_ALL	-23.81403	26.73541	13	188.5	-0.7	4.8	75.4	NaN	NaN	
Masama_2_Sill	PW36_ALL	-23.81453	26.73503	8	183.2	3.5	7.7	53.1	NaN	NaN	
Dibete_Kop_Sill	PW38_ALL	-23.78171	26.56308	7	194.4	0.8	9.4	42.4	NaN	NaN	

The means were all calculated using specimen data that had already been tilt-corrected. In order to present the uncorrected means, we untilt the means using the reversed of the tilt-correction (dip_direction, -dip).

```
In [145]: dec_geo = []
        inc_geo = []
        for n in range(0,len(Intrusions_newBot_VGPs_rnd)):
            dec_g, inc_g = pmag.dotilt(Intrusions_newBot_VGPs_rnd['dec_tc'][n],
                                       Intrusions_newBot_VGPs_rnd['inc_tc'][n],
                                       Intrusions_newBot_VGPs_rnd['dip_direction'][n],
                                       -Intrusions_newBot_VGPs_rnd['dip'][n])
            dec_geo.append(dec_g)
            inc_geo.append(inc_g)
        dec_geo
        inc_geo

        Intrusions_newBot_VGPs_rnd.insert(4,'dec',dec_geo)
        Intrusions_newBot_VGPs_rnd.insert(5,'inc',inc_geo)

        Intrusions_newBot_VGPs_rnd.to_csv('../Data/Umkondo_Sites_new.csv')
```

```
In [146]: All_Umk_VGPs = Intrusions_newBot_VGPs_rnd.append(Old_Data_ALL)
        #Make table to use in publication with only necessary columns and rounded #s
        All_Umk_pub_tbl = pd.DataFrame(data=All_Umk_VGPs)
        All_Umk_pub_tbl=All_Umk_pub_tbl.rename(columns={'site_ID':'sites_used'})
        All_Umk_pub_tbl=All_Umk_pub_tbl.drop('pole_lat_rev',1)
        All_Umk_pub_tbl=All_Umk_pub_tbl.drop('pole_long_rev',1)
        All_Umk_pub_tbl=All_Umk_pub_tbl.drop('paleolatitude',1)
        All_Umk_pub_tbl['pole_lat']=np.round(All_Umk_pub_tbl['pole_lat'], 1)
        All_Umk_pub_tbl['pole_long']=np.round(All_Umk_pub_tbl['pole_long'], 1)
```

```
All_Umk_pub_tbl['inc_tc']=np.round(All_Umk_pub_tbl['inc_tc'], 1)
All_Umk_pub_tbl.to_csv('../Data/Umkondo_Sites_all.csv')
```

```
In [147]: Umk_latex = All_Umk_pub_tbl[['sites_used','site_lat','site_long','n','dec','inc','de
          Umk_latex.site_lat = Umk_latex.site_lat.round(3)
          Umk_latex.site_long = Umk_latex.site_long.round(3)
          Umk_latex.dec = Umk_latex.dec.round(1)
          Umk_latex.inc = Umk_latex.inc.round(1)
          print Umk_latex.to_latex()
```

```
\begin{tabular}{llrrlrrrrrlrr}
\toprule
{} &                     sites\_used &  site\_lat &  site\_long &   n &   dec &   inc &
\midrule
Kgale\_Peak\_Sill          &                     PW1\_ALL and PW2\_AF &   -24.688 &     25.862 &
Rasemong\_Sill            &                                PW5\_ALL &   -24.727 &     25.776 &
Metsemotlhaba\_River\_Sill &                                PW6\_ALL &   -24.547 &     25.809 &
Mabogoapitse\_Hill\_Sill   &                     PW7\_ALL and PW8\_AF &   -24.474 &     25.597 &
Semarule\_Hill\_Sill       &                                PW9\_ALL &   -24.453 &     25.574 &
Rapitsane\_Sill           &                               PW10\_ALL &   -24.420 &     25.585 &
Suping\_Sill              &                      PW11\_ALL and JP15 &   -24.328 &     25.532 &
Mogatelwane 2 Sill       &                               PW15\_ALL &   -24.180 &     25.692 &
Mosolotsane\_1\_Sill       &  PW21\_AF, PW22\_ALL, and JP(22,23,24) &   -22.907 &     26.389 &
Mosolotsane\_5\_Sill       &                               PW23\_ALL &   -22.903 &     26.370 &
Mosolotsane\_4\_Sill       &                                PW24\_AF &   -22.895 &     26.374 &
Mosolotsane\_6\_Sill       &                                PW25\_AF &   -22.896 &     26.367 &
Mosolotsane\_3\_Sill       &                                PW26\_AF &   -22.893 &     26.381 &
Mosolotsane\_2\_Sill       &                                PW27\_AF &   -22.892 &     26.382 &
Shoshong\_Sill            &              PW28\_AF and JP(26,31,33,34) &   -23.005 &     26.484 &
Phage\_Sill               &                                PW29\_AF &   -22.779 &     26.394 &
Moijabana\_Sill           &                                PW30\_AF &   -22.642 &     26.443 &
Mokgware\_Sill            &                      PW31\_ALL and JP30 &   -22.707 &     26.611 &
Sepatamorire\_Sill        &                               PW32\_ALL &   -22.335 &     26.823 &
Palapye\_dike             &                                PW33\_AF &   -22.578 &     27.287 &
Masama\_1\_Sill            &                               PW34\_ALL &   -23.816 &     26.738 &
Masama\_3\_Sill            &                   PW35\_AF and PW37\_ALL &   -23.814 &     26.735 &
Masama\_2\_Sill            &                               PW36\_ALL &   -23.815 &     26.735 &
Dibete\_Kop\_Sill          &                               PW38\_ALL &   -23.782 &     26.563 &
W01\_W02                  &                                W01\_W02 &   -25.480 &     29.450 &
W04                      &                                    W04 &   -25.750 &     29.450 &   12
W05                      &                                    W05 &   -25.760 &     29.480 &   11
W08\_W09                  &                                W08\_W09 &   -25.620 &     29.100 &
VF1\_VF2                  &                                VF1\_VF2 &   -25.800 &     27.500 &
TG-S-series              &                            TG-S-series &   -24.200 &     31.400 &  120
TG-N-series              &                            TG-N-series &   -23.200 &     31.200 &   13
```

```
JP19                    &                                 JP19 &   -24.230 &     25.640 &    5
J\_M7                   &                                  J\_M7 &   -24.330 &     26.130 &
J\_M8                   &                                  J\_M8 &   -24.230 &     25.870 &
J\_M3                   &                                  J\_M3 &   -23.000 &     26.410 &
J\_M10                  &                                 J\_M10 &   -22.920 &     29.930 &
J\_M12                  &                                 J\_M12 &   -26.900 &     28.530 &
J\_M13                  &                                 J\_M13 &   -25.700 &     28.530 &
M\_O\_B                  &                                M\_O\_B &   -18.100 &     32.900 &
M\_O\_D                  &                                M\_O\_D &   -18.450 &     32.760 &
M\_O\_E                  &                                M\_O\_E &   -19.530 &     32.630 &
M\_O\_F                  &                                M\_O\_F &   -19.600 &     32.800 &
M\_O\_H                  &                                M\_O\_H &   -19.850 &     32.950 &
M\_O\_J                  &                                M\_O\_J &   -20.530 &     32.660 &
WD1                     &                                  WD1 &   -23.810 &     28.740 &    9
WD8                     &                                  WD8 &   -24.280 &     28.710 &   12
WD17                    &                                 WD17 &   -23.150 &     28.750 &   10
WD19                    &                                 WD19 &   -23.160 &     26.680 &   10
WD25                    &                                 WD25 &   -23.420 &     28.650 &    8
WD26                    &                                 WD26 &   -23.950 &     28.390 &   13
WD32                    &                                 WD32 &   -24.140 &     27.410 &    6
WD33                    &                                 WD33 &   -24.050 &     27.320 &   10
WD34                    &                                 WD34 &   -23.840 &     26.930 &    7
WRD4                    &                                 WRD4 &   -25.660 &     29.160 &    5
WRD5                    &                                 WRD5 &   -25.880 &     29.030 &    5
WRD6                    &                                 WRD6 &   -25.820 &     28.950 &    8
WRD7                    &                                 WRD7 &   -25.710 &     28.710 &    8
Wil\_1                  &                                 Wil\_1 &   -17.900 &     31.500 &
Wil\_2                  &                                 Wil\_2 &   -17.400 &     30.100 &
\bottomrule
\end{tabular}

/Users/Laurentia/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-packages/pandas/core/ge
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing
self[name] = value
```

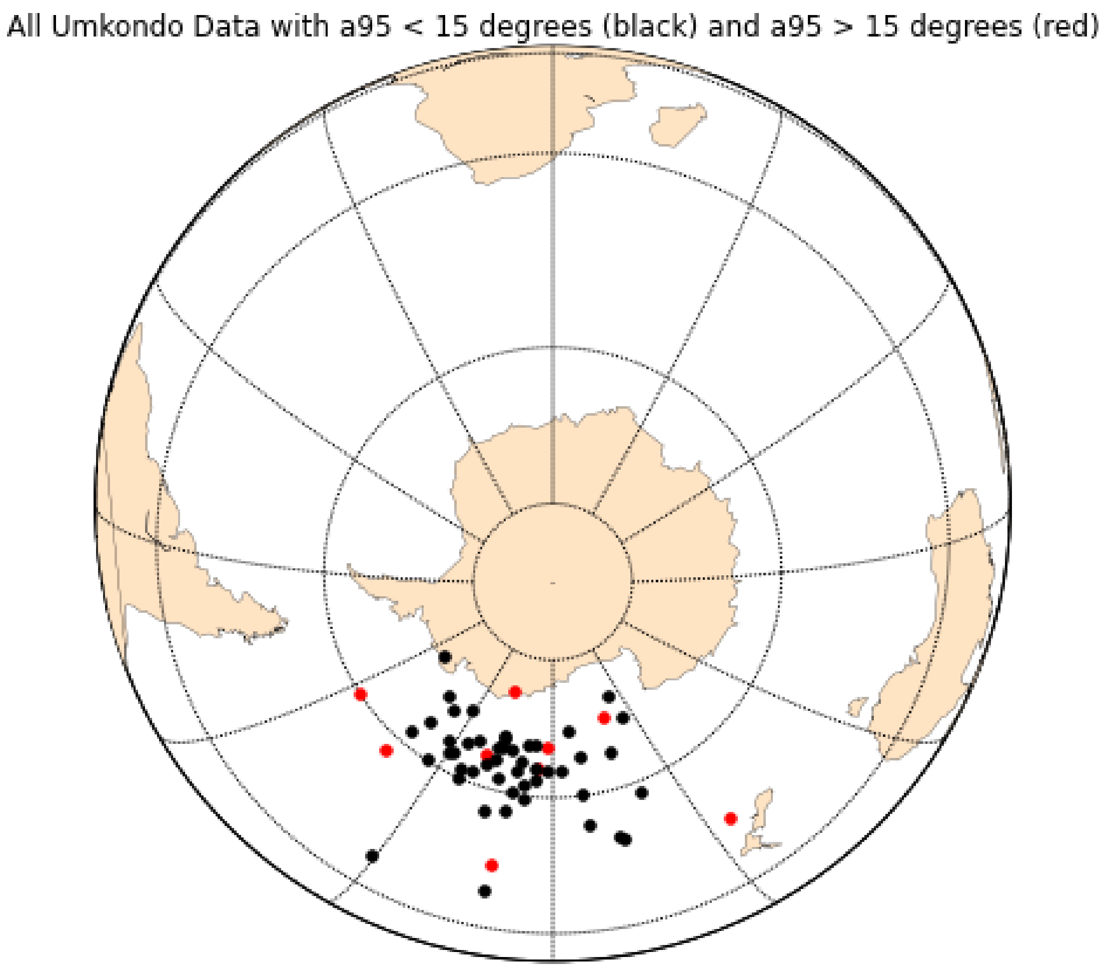
```
In [148]: # a95 filter - 15 degrees
          # created two dataframes applying this filter
          Umk_VGPs_a95_filtered = All_Umk_VGPs.ix[All_Umk_VGPs['a_95']<15]
          Umk_VGPs_high_a95 = All_Umk_VGPs.ix[All_Umk_VGPs['a_95']>15]

          Umk_VGPs_a95_filtered.to_csv('../Data/Umkondo_Site_Means_a95_filtered.csv')
          Umk_VGPs_a95_filtered
```

Out[148]:

	a_95	date	date_error	dec	dec_tc	dip	dip_direction	inc	inc_tc	k	n	paleolatitude
Kgale_Peak_Sill	6.3	1108.0	0.9	189.728559	189.9	6	224.6	5.329220	0.40	48.8	12	0.200002
Rasemong_Sill	8.1	NaN	NaN	14.400000	14.4	0	0.0	-18.600000	-18.60	48.0	8	-9.551602
Metsemotlhaba_River_Sill	14.4	NaN	NaN	180.600000	180.6	0	0.0	-2.700000	-2.70	18.5	7	-1.350750
Mabogoapitse_Hill_Sill	9.1	NaN	NaN	184.600000	184.6	0	0.0	5.100000	5.10	33.2	9	2.555061
Semarule_Hill_Sill	9.1	NaN	NaN	186.800000	186.8	0	0.0	3.800000	3.80	72.2	5	1.902092
Rapitsane_Sill	8.5	NaN	NaN	197.931464	197.7	7	330.0	-4.903842	-0.20	43.3	8	-0.100000
Suping_Sill	8.4	NaN	NaN	188.871011	187.2	10	281.2	-9.753488	-9.20	20.2	16	-4.629842
Mogatelwane 2 Sill	14.7	NaN	NaN	193.629006	193.5	4	260.3	-1.219552	-2.80	21.6	6	-1.400836
Mosolotsane_1_Sill	4.6	1109.3	0.6	186.853869	186.1	10	262.9	-3.246794	-5.60	36.9	27	-2.806703
Mosolotsane_5_Sill	14.2	NaN	NaN	189.598220	188.5	10	262.9	-5.111960	-7.90	19.1	7	-3.968863
Mosolotsane_4_Sill	7.9	NaN	NaN	185.443022	185.2	10	262.9	-0.343753	-2.50	50.3	8	-1.250595
Mosolotsane_6_Sill	9.0	NaN	NaN	188.939129	191.2	10	262.9	14.759440	11.80	72.7	5	5.963231
Mosolotsane_2_Sill	5.6	NaN	NaN	187.048911	187.6	10	262.9	4.497009	2.00	97.5	8	1.000305
Shoshong_Sill	3.1	1109.3	0.4	191.500000	191.5	0	0.0	-5.400000	-5.40	65.2	33	-2.706009
Phage_Sill	7.8	NaN	NaN	193.930454	194.0	10	270.0	1.619116	-0.80	50.9	8	-0.400019
Mokgware_Sill	6.5	1112.0	0.5	199.208120	199.0	4	90.1	2.496984	3.80	42.2	13	1.902092
Sepatamorire_Sill	8.3	NaN	NaN	194.100000	194.1	0	0.0	1.500000	1.50	45.6	8	0.750129
Palapye_dike	11.4	NaN	NaN	172.504631	173.6	9	312.1	7.096386	13.90	29.0	7	7.053782
Masama_1_Sill	8.8	NaN	NaN	169.629937	170.5	8	10.4	-21.103100	-13.60	40.3	8	-6.897145
Masama_3_Sill	4.8	NaN	NaN	188.478042	188.5	8	10.4	-8.695565	-0.70	75.4	13	-0.350013
Masama_2_Sill	7.7	NaN	NaN	183.191783	183.2	8	10.4	-4.437048	3.50	53.1	8	1.751634
Dibete_Kop_Sill	9.4	NaN	NaN	194.400000	194.4	0	0.0	0.800000	0.80	42.4	7	0.400019
W01_W02	6.2	NaN	NaN	NaN	175.6	NaN	NaN	NaN	-18.40	22.6	25	-9.443434
W04	4.9	NaN	NaN	NaN	171.5	NaN	NaN	NaN	-22.30	80.8	12	-11.588697
W05	7.5	NaN	NaN	NaN	176.4	NaN	NaN	NaN	-7.80	38.0	11	-3.918154
W08_W09	1.9	NaN	NaN	NaN	192.8	NaN	NaN	NaN	15.90	297.2	20	8.106047
VF1_VF2	3.1	1108.6	1.2	NaN	7.2	NaN	NaN	NaN	-6.80	103.2	21	-3.412015
TG-S-series	5.7	1111.5	0.4	NaN	186.3	NaN	NaN	NaN	2.87	7.2	120	1.435901
TG-N-series	6.5	NaN	NaN	NaN	182.8	NaN	NaN	NaN	-14.73	41.9	13	-7.488730
JP19	14.9	NaN	NaN	NaN	188.2	NaN	NaN	NaN	-15.45	27.2	5	-7.868014
J_M7	5.2	NaN	NaN	NaN	193.5	NaN	NaN	NaN	-5.50	165.0	6	-2.800000
J_M3	2.0	NaN	NaN	NaN	190.5	NaN	NaN	NaN	4.00	796.0	8	2.000000
J_M10	9.5	NaN	NaN	NaN	194.0	NaN	NaN	NaN	24.00	66.5	5	12.600000
J_M12	3.9	1108.5	0.8	NaN	16.0	NaN	NaN	NaN	-14.50	292.0	6	-7.400000
J_M13	5.7	NaN	NaN	NaN	183.0	NaN	NaN	NaN	-3.00	73.5	10	-1.500000
M_O_B	4.5	NaN	NaN	NaN	171.5	NaN	NaN	NaN	-10.00	267.0	5	-5.000000
M_O_D	14.0	NaN	NaN	NaN	168.0	NaN	NaN	NaN	-5.50	12.6	10	-2.800000
M_O_E	5.0	NaN	NaN	NaN	185.0	NaN	NaN	NaN	-3.50	92.0	10	-1.800000
M_O_F	4.0	NaN	NaN	NaN	179.5	NaN	NaN	NaN	-13.00	206.0	8	-6.600000
M_O_H	10.5	NaN	NaN	NaN	185.0	NaN	NaN	NaN	-2.50	21.0	10	-1.300000
WD1	13.4	NaN	NaN	NaN	184.0	NaN	NaN	NaN	-8.50	15.7	9	-4.300000
WD8	9.6	NaN	NaN	NaN	171.4	NaN	NaN	NaN	-26.30	21.5	12	-13.900000
WD17	9.5	NaN	NaN	NaN	189.5	NaN	NaN	NaN	-18.80	26.8	10	-9.700000
WD19	12.9	NaN	NaN	NaN	190.5	NaN	NaN	NaN	-43.50	15.1	10	-25.400000
WD26	9.0	NaN	NaN	NaN	171.7	NaN	NaN	NaN	10.60	22.1	13	5.300000
WD33	14.9	NaN	NaN	NaN	206.9	NaN	NaN	NaN	-36.20	11.5	10	-20.100000
WRD4	8.2	NaN	NaN	NaN	178.1	NaN	NaN	NaN	11.10	NaN	5	5.600000
Wil_1	7.9	NaN	NaN	NaN	181.4	NaN	NaN	NaN	-15.40	NaN	5	7.800000
Wil_2	11.6	NaN	NaN	NaN	10.6	NaN	NaN	NaN	9.30	NaN	7	4.700000

```
In [149]: #create basemap for VGP plot
          plt.figure(figsize=(7, 7))
          m1 = Basemap(projection='ortho',lat_0=-80,lon_0=30,resolution='c',
                       area_thresh=50000)
          m1.drawcoastlines(linewidth=0.25)
          m1.fillcontinents(color='bisque',lake_color='white',zorder=1)
          m1.drawmapboundary(fill_color='white')
          m1.drawmeridians(np.arange(0,360,30))
          m1.drawparallels(np.arange(-90,90,30))

          #reverse selected poles so all in same polarity - South-seeking
          for n in range(len(Umk_VGPs_high_a95)):
              if All_Umk_VGPs['pole_lat'][n] < 0:
                  IPmag.vgpplot(m1,Umk_VGPs_high_a95['pole_long'][n],
                                Umk_VGPs_high_a95['pole_lat'][n],
                                label=Umk_VGPs_high_a95.index[n],color='r')
```

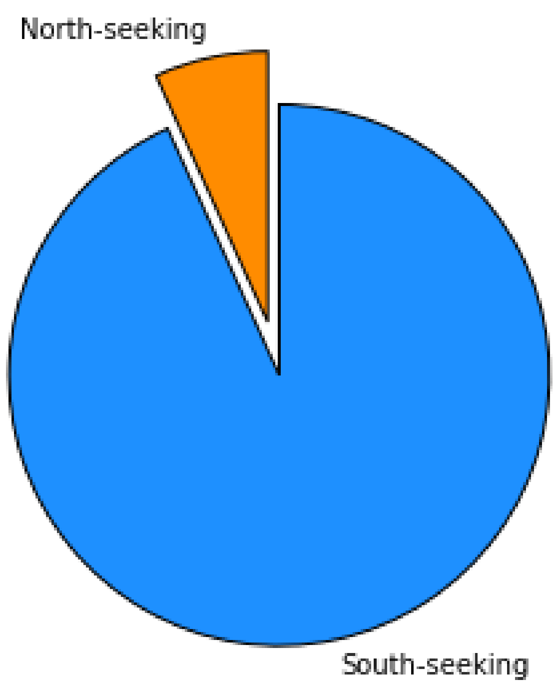
```
    else: IPmag.vgpplot(m1,Umk_VGPs_high_a95['pole_long_rev'][n],
                        Umk_VGPs_high_a95['pole_lat_rev'][n],
                        label=All_Umk_VGPs.index[n],color='r')
for n in range(len(Umk_VGPs_a95_filtered)):
    if Umk_VGPs_a95_filtered['pole_lat'][n] < 0:
        IPmag.vgpplot(m1,Umk_VGPs_a95_filtered['pole_long'][n],
                      Umk_VGPs_a95_filtered['pole_lat'][n],
                      label=Umk_VGPs_a95_filtered.index[n])
    else: IPmag.vgpplot(m1,Umk_VGPs_a95_filtered['pole_long_rev'][n],
                        Umk_VGPs_a95_filtered['pole_lat_rev'][n],
                        label=Umk_VGPs_a95_filtered.index[n])

plt.title('All Umkondo Data with a95 < 15 degrees (black) and a95 > 15 degrees (red)')
plt.show()
```



4.1 Polarity of VGPs

Here are the polarities of the VGPs considering them unfiltered by α_{95}.

```
In [150]: labels = 'North-seeking', 'South-seeking'
          sizes = [len(All_Umk_VGPs.ix[All_Umk_VGPs['pole_lat'] > 0])
                   ,len(All_Umk_VGPs.ix[All_Umk_VGPs['pole_lat'] < 0])]
          colors = ['darkorange','dodgerblue']
          explode = (0.2, 0)
          plt.pie(sizes, explode=explode, labels=labels, colors=colors, shadow=False,
                  startangle=90)
          plt.axis('equal')
          plt.show()
          print 'The total number of VGPs in the compilation is:'
          print len(All_Umk_VGPs)
          print 'This many have north seeking declinations:'
          print len(All_Umk_VGPs.ix[All_Umk_VGPs['pole_lat'] > 0])
          print str((4.0/59.0) * 100) + ' percent of the total'
          print 'This many have south seeking declinations:'
          print len(All_Umk_VGPs.ix[All_Umk_VGPs['pole_lat'] < 0])
```




```
The total number of VGPs in the compilation is:
```

```
59
This many have north seeking declinations:
4
6.77966101695 percent of the total
This many have south seeking declinations:
55
```

4.1.1 Calculating poles by polarity

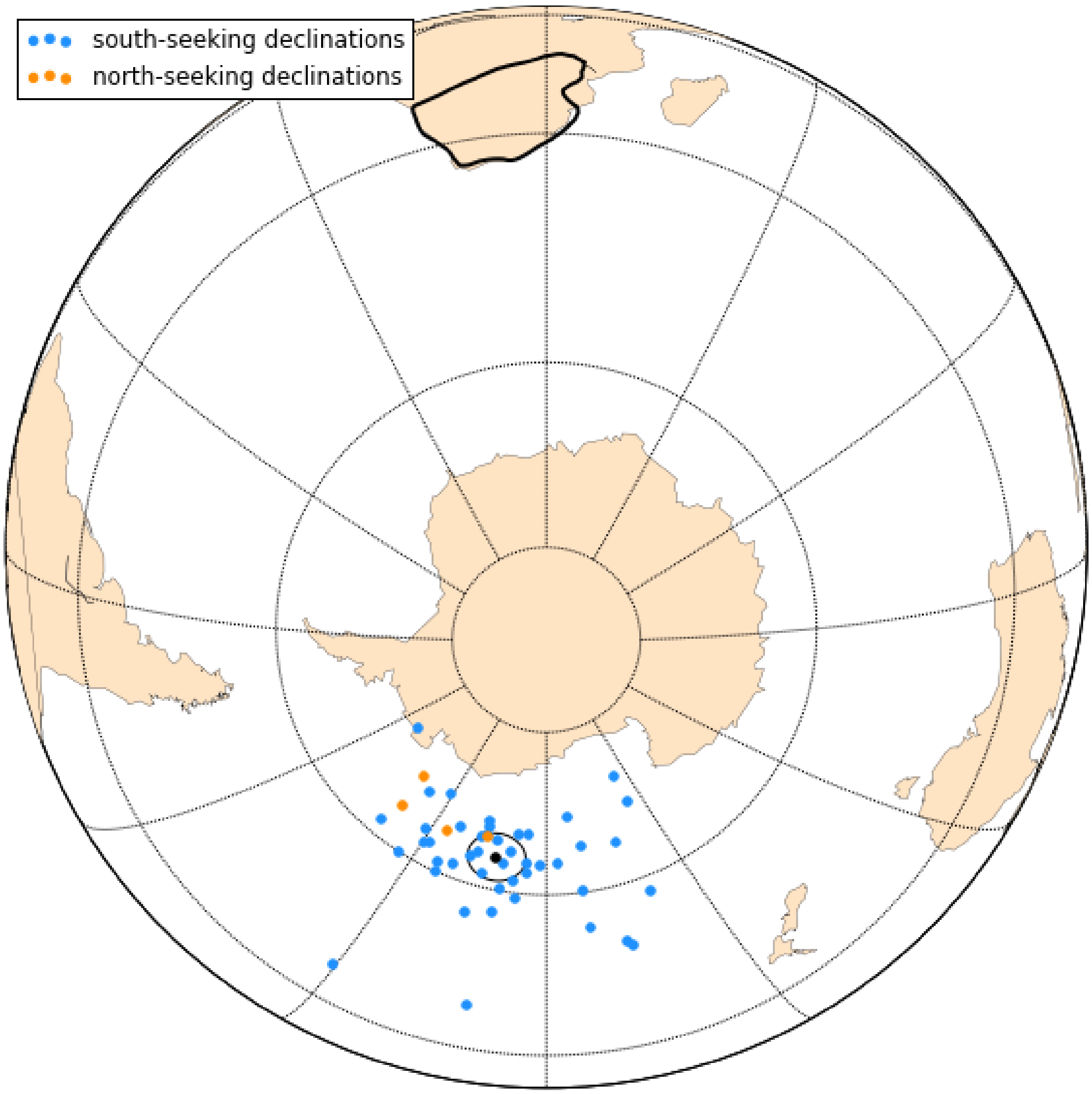
Using the α_{95} filtered data, we can calculate a grand mean pole and then distinct poles grouped by polarity.

```
In [151]: Umk_VGPs_a95_filtered_N = Umk_VGPs_a95_filtered.ix[Umk_VGPs_a95_filtered['pole_lat']
          Umk_VGPs_a95_filtered_S = Umk_VGPs_a95_filtered.ix[Umk_VGPs_a95_filtered['pole_lat'] < 0

          Umk_VGP_N_block = IPmag.make_di_block(Umk_VGPs_a95_filtered_N['pole_long_rev'],
                                                Umk_VGPs_a95_filtered_N['pole_lat_rev'])
          Umk_direction_N_block = IPmag.make_di_block(Umk_VGPs_a95_filtered_N['dec_tc']+180,
                                                Umk_VGPs_a95_filtered_N['inc_tc']*-1)

          Umk_VGP_N_mean = pmag.fisher_mean(Umk_VGP_N_block)

          Umk_VGP_S_block = IPmag.make_di_block(Umk_VGPs_a95_filtered_S['pole_long'],
                                                Umk_VGPs_a95_filtered_S['pole_lat'])

          Umk_VGP_S_mean = pmag.fisher_mean(Umk_VGP_S_block)

          Umk_VGP_all_block = Umk_VGP_N_block+Umk_VGP_S_block
          Umk_VGP_all_mean = pmag.fisher_mean(Umk_VGP_all_block)

          print 'The fisher mean parameters for the Umkondo Grand Mean pole are: '
          print str(Umk_VGP_all_mean)
          print 'The fisher mean parameters for the Umkondo north-seeking directions are: '
          print str(Umk_VGP_N_mean)
          print 'The fisher mean parameters for the Umkondo south-seeking directions are: '
          print str(Umk_VGP_S_mean)
```

```
The fisher mean parameters for the Umkondo Grand Mean pole are:
{'csd': 10.432348476278413, 'k': 60.284528236779231, 'n': 49, 'r': 48.203775804440724, 'alpha9
The fisher mean parameters for the Umkondo north-seeking directions are:
{'csd': 4.9406124554534196, 'k': 268.78712371113301, 'n': 4, 'r': 3.9888387510585361, 'alpha95
The fisher mean parameters for the Umkondo south-seeking directions are:
{'csd': 10.515958705513198, 'k': 59.329719310644279, 'n': 45, 'r': 44.258381793960282, 'alpha9
```

4.1.2 Comparison between the two polarities

```
In [152]: #create basemap for VGP plot with all poles in same polarity
          plt.figure(figsize=(10, 10))
          m1 = Basemap(projection='ortho',lat_0=-80,lon_0=30,resolution='c',
                       area_thresh=50000)
          m1.drawcoastlines(linewidth=0.25)
          m1.fillcontinents(color='bisque',lake_color='white',zorder=1)
          m1.drawmapboundary(fill_color='white')
          m1.drawmeridians(np.arange(0,360,30))
          m1.drawparallels(np.arange(-90,90,30))
          m1.readshapefile('../Data/Kalahari_Outline/reconstructed_0Ma_PLATE_ID__7702','Kalahari',
                           drawbounds=True,zorder=10,linewidth=2)
          IPmag.vgpplot(m1,Umk_VGPs_a95_filtered_S['pole_long'].tolist(),
                        Umk_VGPs_a95_filtered_S['pole_lat'].tolist(),
                        color='dodgerblue',label='south-seeking declinations')
          IPmag.vgpplot(m1,(180.+Umk_VGPs_a95_filtered_N['pole_long']).tolist(),
                        (-1*Umk_VGPs_a95_filtered_N['pole_lat']).tolist(),
                        color='darkorange',label='north-seeking declinations')
          IPmag.poleplot(m1,222.3,-64.3,3.1,color='k')
          plt.legend(loc='upper left')
          plt.savefig('Code_Output/Umkondo_mean_pole.pdf')
```



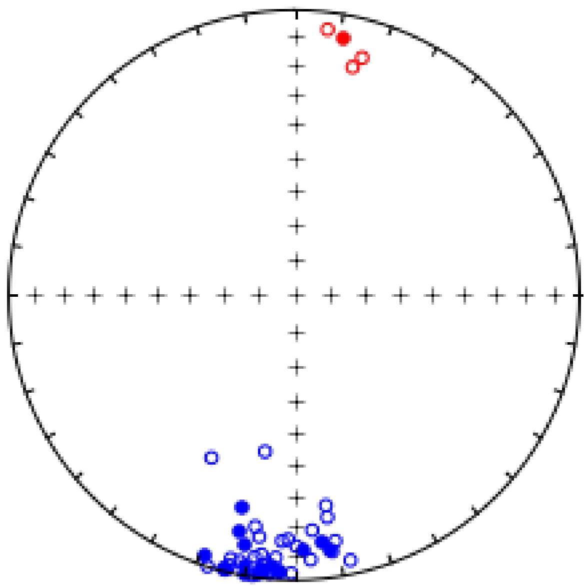

```
In [153]: IPmag.watson_common_mean(Umk_VGP_N_block,Umk_VGP_S_block)
```

```
Results of Watson V test:

Watson's V:           18.1
Critical value of V:  7.4
"Fail": Since V is greater than Vcrit, the two means can
be distinguished at the 95% confidence level.

M&M1990 classification:
```

```
Angle between data set means: 8.8
Critical angle for M&M1990:   5.6
```

4.1.3 Watson common mean test of R and N Umkondo data in directional space

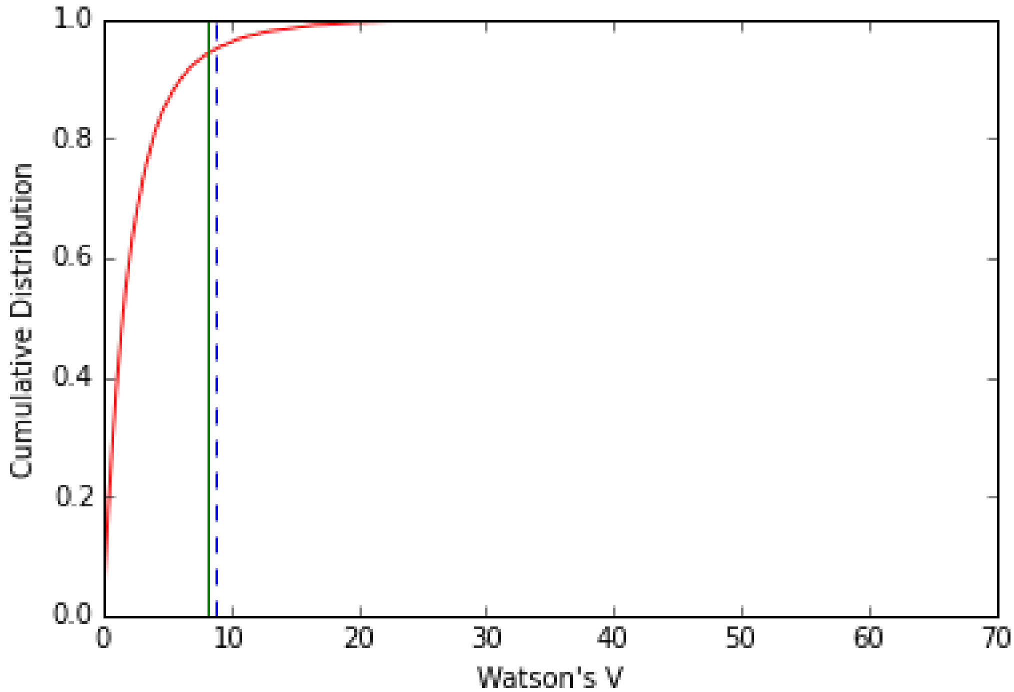
```
In [154]: Umk_directions_S = IPmag.make_di_block(Umk_VGPs_a95_filtered_S['dec_tc'],
                                                 Umk_VGPs_a95_filtered_S['inc_tc'])

          Umk_directions_N = IPmag.make_di_block(Umk_VGPs_a95_filtered_N['dec_tc'],
                                                 Umk_VGPs_a95_filtered_N['inc_tc'])

          #reverse the polarity of data so that tests can be performed;...
          #...module definitions require the mean populations be in the same polarity.
          Umk_directions_N_flipped = IPmag.iflip(Umk_directions_N)

          #dataset used to run the bootstrap and Watson tests for common means
          fignum = 1
          plt.figure(num=fignum,figsize=(5,5))
          IPmag.iplotNET(1)
          IPmag.iplotDI(Umk_directions_N,color='red')
          IPmag.iplotDI(Umk_directions_S,color='blue')
```



```
In [155]: IPmag.watson_common_mean(Umk_directions_S, Umk_directions_N_flipped, NumSims=10000,
                                   plot='yes')
```

```
Results of Watson V test:

Watson's V:           8.1
Critical value of V:  8.9
"Pass": Since V is less than Vcrit, the null hypothesis
that the two populations are drawn from distributions
that share a common mean direction can not be rejected.

M&M1990 classification:

Angle between data set means: 13.9
Critical angle for M&M1990:   14.5
```

```
The McFadden and McElhinny (1990) classification for
this test is: 'C'
```

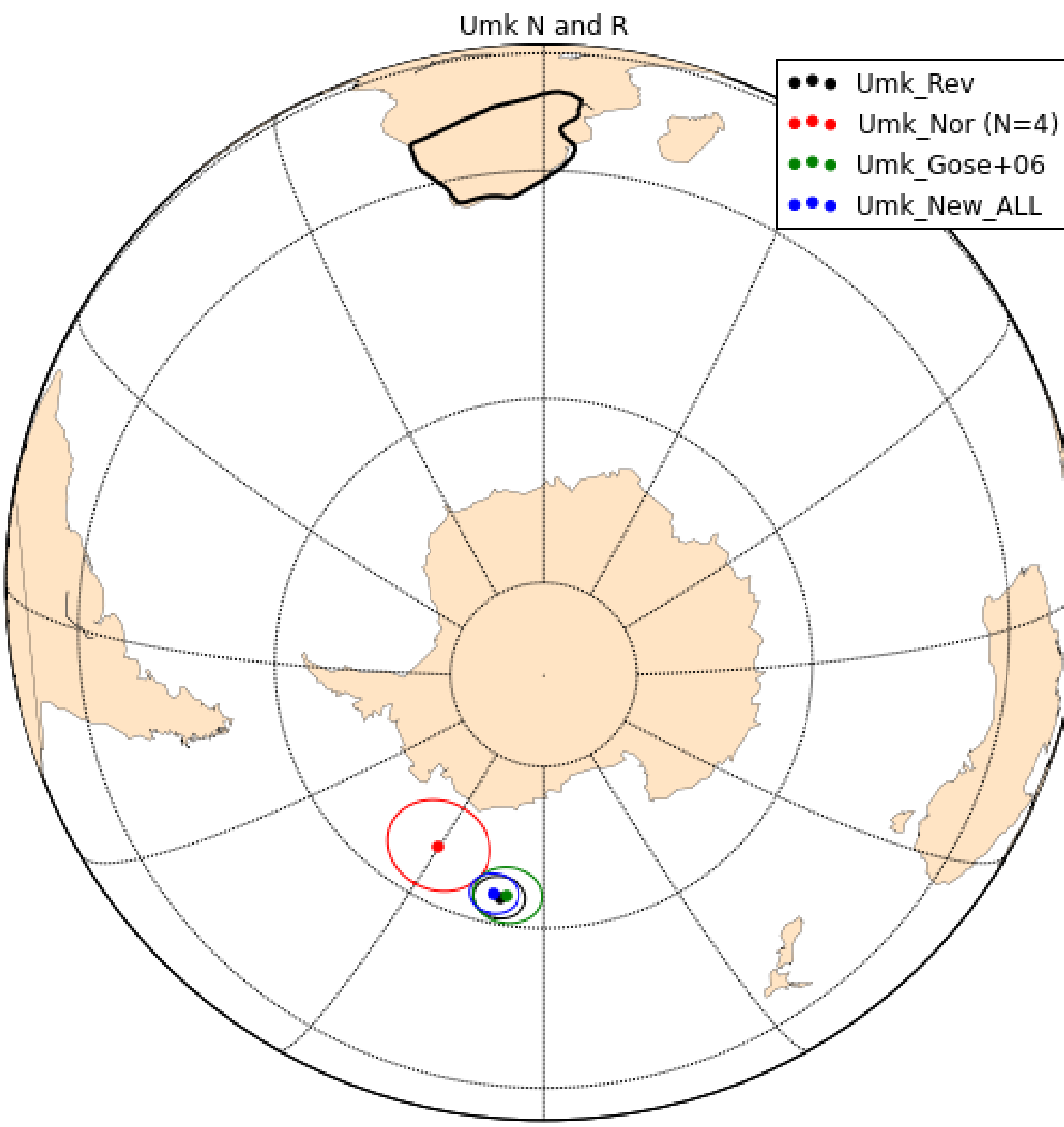


In directional space, the north seeking and south seeking directions are consistent with sharing a common mean. In VGP space they are not. However, given that there are only 4 north-seeking directions, and that the north-seeking directions are therefore unlikely to average out secular variation, neither of these statistical tests comparing the two polarities are not particularly robust. The populations could be identical with the north-seeking polarity being poorly sampled. This analysis will be interesting to revisit if future work develops data from additional north-seeking sites within the LIP.

4.1.4 Plotting up mean poles and generating Table 2 of main manuscript

```
In [156]: #create basemap for pole plot
      plt.figure(figsize=(9, 9))
      m1 = Basemap(projection='ortho',lat_0=-80,lon_0=30,resolution='c',
                   area_thresh=50000)
      m1.drawcoastlines(linewidth=0.25)
      m1.fillcontinents(color='bisque',lake_color='white',zorder=1)
      m1.readshapefile('../Data/Kalahari_Outline/reconstructed_0Ma_PLATE_ID__7702','Kalahari',
                       drawbounds=True,zorder=10,linewidth=2)
      m1.drawmapboundary(fill_color='white')
```

```
m1.drawmeridians(np.arange(0,360,30))
m1.drawparallels(np.arange(-90,90,30))

IPmag.poleplot(m1,Umk_VGP_S_mean['dec'],Umk_VGP_S_mean['inc'],Umk_VGP_S_mean['alpha95'],
               label='Umk_Rev')
IPmag.poleplot(m1,Umk_VGP_N_mean['dec'],Umk_VGP_N_mean['inc'],Umk_VGP_N_mean['alpha95'],
               label='Umk_Nor (N=4)',color='r')
IPmag.poleplot(m1,218.8,-64,3.7,label='Umk_Gose+06',color='g')
IPmag.poleplot(m1,Umk_VGP_all_mean['dec'],Umk_VGP_all_mean['inc'],
               Umk_VGP_all_mean['alpha95'],label='Umk_New_ALL',color='b')

plt.title('Umk N and R')
plt.legend()
plt.show()
```




```
In [157]: Umk_poles=[]
       Umk_poles.append(Umk_VGP_all_mean)
       Umk_poles.append(Umk_VGP_N_mean)
       Umk_poles.append(Umk_VGP_S_mean)

       UmkPoles = pd.DataFrame(data=Umk_poles, index=('Umk_New_ALL','Umk_North_Seeking',
                                                      'Umk_South_Seeking'))
       UmkPoles.loc['Gose et al. (2006)'] = pd.Series({'alpha95':'3.7',
                                                  'inc':-64,
```

```
                                            'dec':218.8,
                                            'n':10,
                                            'k':172.2,
                                            'r':9.948})
        #Insert other columns for complete table
        # Tidy up table
        UmkPoles = UmkPoles.rename(columns = {'csd':'CSD'})
        UmkPoles = UmkPoles.rename(columns = {'alpha95':'A_95'})
        UmkPoles = UmkPoles.rename(columns = {'dec':'Pole_Long'})
        UmkPoles = UmkPoles.rename(columns = {'inc':'Pole_Lat'})
        UmkPoles = UmkPoles.rename(columns = {'k':'K'})
        UmkPoles = UmkPoles.rename(columns = {'n':'N'})
        #reorder columns
        UmkPoles = UmkPoles[['Pole_Lat','Pole_Long','A_95','K','CSD','N','r']]
        UmkPoles
```

Out[157]:

	Pole_Lat	Pole_Long	A_95	K	CSD	N	r
Umk_New_ALL	-63.958432	222.103677	2.642991	60.284528	10.432348	49	48.203776
Umk_North_Seeking	-67.072651	240.263175	5.614394	268.787124	4.940612	4	3.988839
Umk_South_Seeking	-63.580944	220.694431	2.784403	59.329719	10.515959	45	44.258382
Gose et al. (2006)	-64.000000	218.800000	3.7	172.200000	NaN	10	9.948000

```
In [158]: print UmkPoles.to_latex()
```

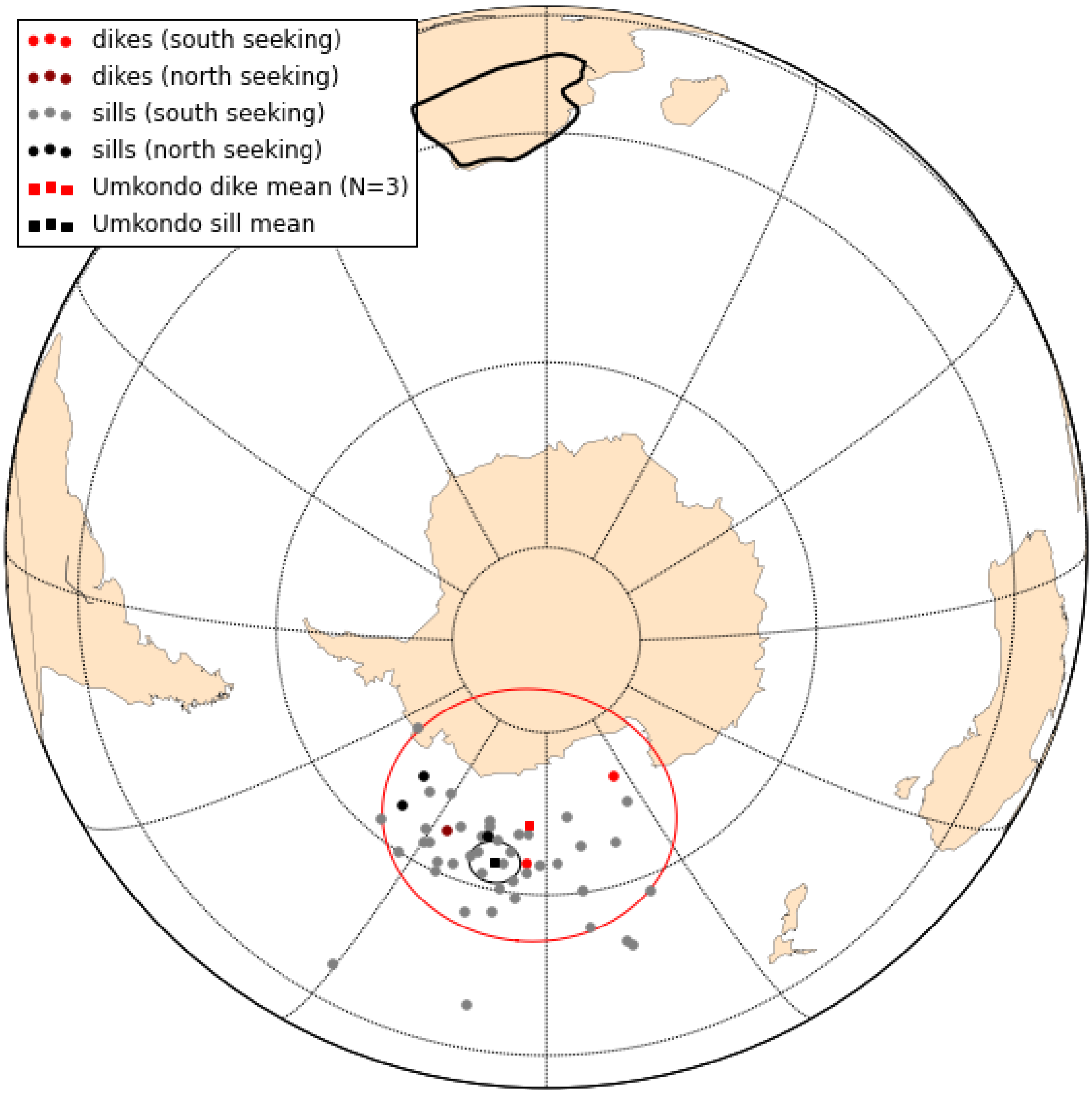
```
\begin{tabular}{lrrlrrrr}
\toprule
{} &   Pole\_Lat &   Pole\_Long &      A\_95 &            K &         CSD &   N &            r \\
\midrule
Umk\_New\_ALL        & -63.958432 &  222.103677 &  2.642991 &   60.284528 &  10.432348 &  49 &
Umk\_North\_Seeking  & -67.072651 &  240.263175 &  5.614394 &  268.787124 &   4.940612 &   4 &
Umk\_South\_Seeking  & -63.580944 &  220.694431 &  2.784403 &   59.329719 &  10.515959 &  45 &
Gose et al. (2006) & -64.000000 &  218.800000 &       3.7 &  172.200000 &        NaN &  10 &
\bottomrule
\end{tabular}
```

4.1.5 Comparing VGPs between sills and dikes

All sites included in the compilation are sills with the exception of M-O-J which is a lava flow and Wil-1, Wil-2 and Palapye-dike which are dikes. Prompted by a reviewer's suggestion that there may be a statistical difference between the directions recorded by dikes and sills in the LIP these populations are considered in comparision to one another below.

```
In [159]: Umk_sills_S = Umk_VGPs_a95_filtered_S.drop(['Wil_1', 'Palapye_dike'])
          Umk_sills_N = Umk_VGPs_a95_filtered_N.drop(['Wil_2'])
```

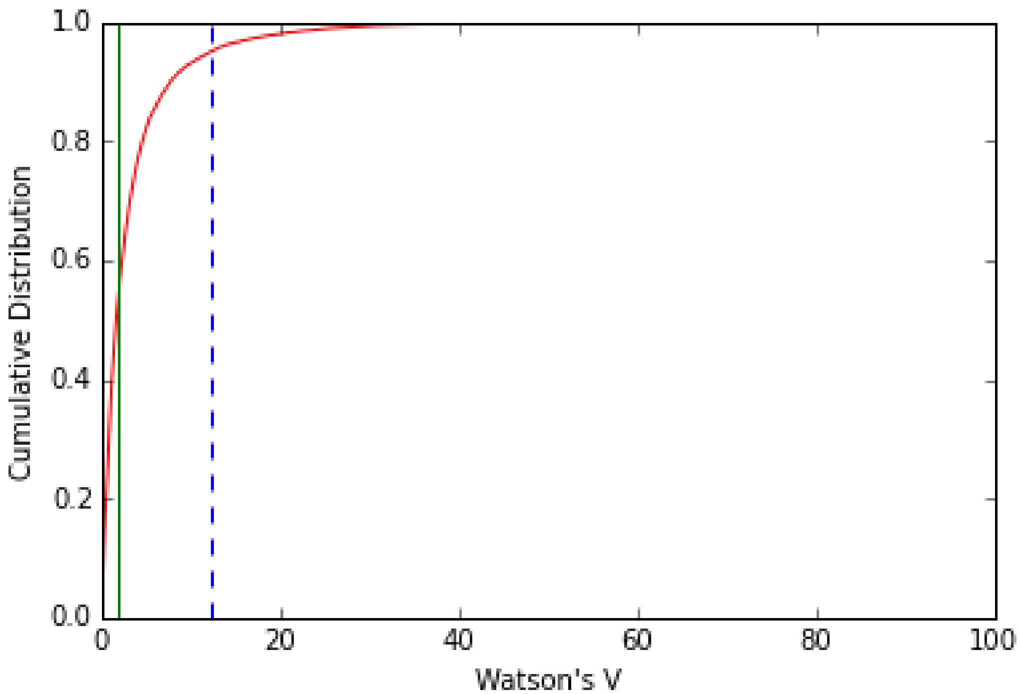
```
Umk_dikes_S = Umk_VGPs_a95_filtered_S.ix[['Wil_1','Palapye_dike']]
Umk_dikes_N = Umk_VGPs_a95_filtered_N.ix[['Wil_2']]

Umk_dikes_VGP = IPmag.make_di_block(Umk_dikes_S['pole_long'],
                                    Umk_dikes_S['pole_lat'])+IPmag.make_di_block(Umk_dike
                                                                                 Umk_dil

Umk_sills_VGP = IPmag.make_di_block(Umk_sills_S['pole_long'],
                                    Umk_sills_S['pole_lat'])+IPmag.make_di_block(Umk_sil
                                                                                 Umk_si

Umk_dikes_mean = pmag.fisher_mean(Umk_dikes_VGP)
Umk_sills_mean = pmag.fisher_mean(Umk_sills_VGP)

plt.figure(figsize=(10, 10))
m1 = Basemap(projection='ortho',lat_0=-80,lon_0=30,resolution='c',
             area_thresh=50000)
m1.drawcoastlines(linewidth=0.25)
m1.fillcontinents(color='bisque',lake_color='white',zorder=1)
m1.drawmapboundary(fill_color='white')
m1.drawmeridians(np.arange(0,360,30))
m1.drawparallels(np.arange(-90,90,30))
m1.readshapefile('../Data/Kalahari_Outline/reconstructed_0Ma_PLATE_ID__7702','Kalahari',
                 drawbounds=True,zorder=10,linewidth=2)
IPmag.vgpplot(m1,Umk_dikes_S['pole_long'].tolist(),Umk_dikes_S['pole_lat'].tolist(),
              color='red',label='dikes (south seeking)')
IPmag.vgpplot(m1,Umk_dikes_N['pole_long_rev'].tolist(),Umk_dikes_N['pole_lat_rev'].tolis
              color='darkred',label='dikes (north seeking)')
IPmag.vgpplot(m1,Umk_sills_S['pole_long'].tolist(),Umk_sills_S['pole_lat'].tolist(),
              color='grey',label='sills (south seeking)')
IPmag.vgpplot(m1,Umk_sills_N['pole_long_rev'].tolist(),Umk_sills_N['pole_lat_rev'].tolis
              color='black',label='sills (north seeking)')
IPmag.poleplot(m1,Umk_dikes_mean['dec'],Umk_dikes_mean['inc'],Umk_dikes_mean['alpha95'],
               label='Umkondo dike mean (N=3)',color='r',marker='s')
IPmag.poleplot(m1,Umk_sills_mean['dec'],Umk_sills_mean['inc'],Umk_sills_mean['alpha95'],
               label='Umkondo sill mean',marker='s')
plt.legend(loc='upper left')
plt.show()
```




```
In [160]: IPmag.watson_common_mean(Umk_dikes_VGP, Umk_sills_VGP, NumSims=10000,
                                   plot='yes')
```

```
Results of Watson V test:

Watson's V:           1.9
Critical value of V:  12.2
"Pass": Since V is less than Vcrit, the null hypothesis
that the two populations are drawn from distributions
that share a common mean direction can not be rejected.
```

```
M&M1990 classification:

Angle between data set means: 6.0
Critical angle for M&M1990:   15.3
The McFadden and McElhinny (1990) classification for
this test is: 'C'
```



The VGPs from the dikes (N=3) and the sills (N=47) pass the Watson test for a common mean. However, given that there are only three VGPs from dikes this test is not particularly robust.

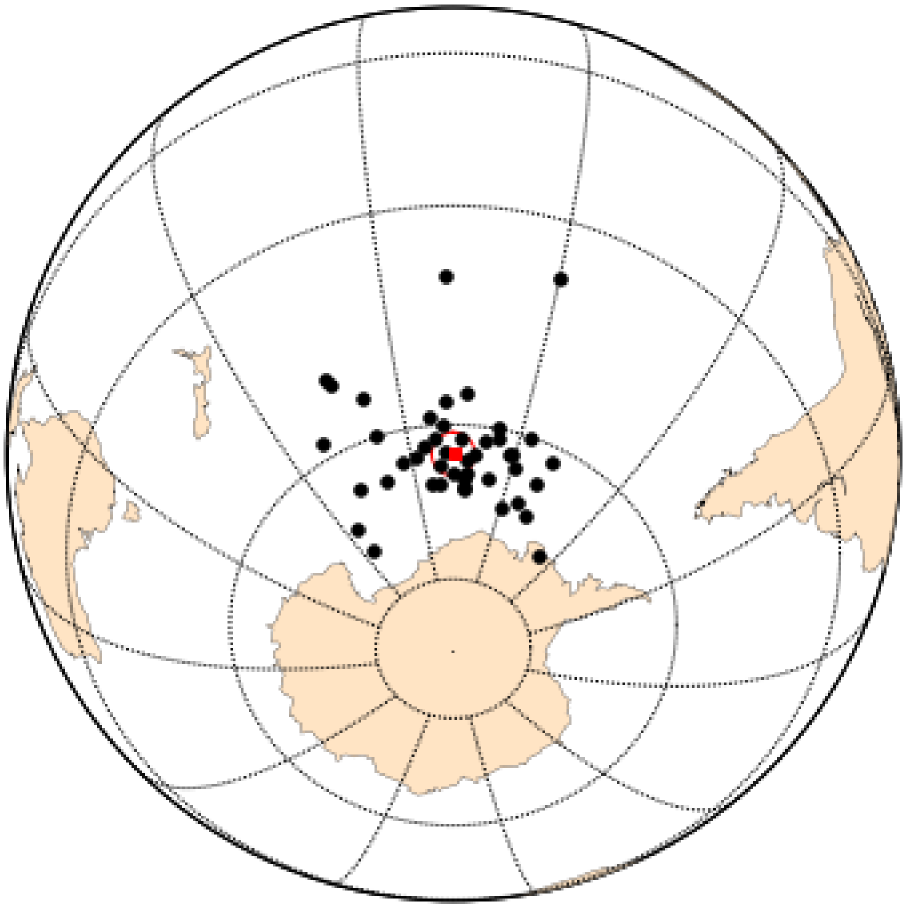
4.1.6 Calculate scatter values, S_w, for each VGP used in this study, according to Biggin (2008)

Add necessary values to master VGP table (above: 'All_Umk_VGPs_edit'), then calculate S_b for the Umkondo pole.

```
In [161]: #First restructured table so sb_vgp_calc func recognizes column names etc.
          #All_Umk_VGPs_S will be used *just* for calculating the scatter
          All_Umk_VGPs_scatter = pd.DataFrame(data=Umk_VGPs_a95_filtered)
          for n in range(len(All_Umk_VGPs_scatter)):
```

```
        if All_Umk_VGPs_scatter['pole_lat'][n] > 0:
            All_Umk_VGPs_scatter['pole_lat'][n] = np.float(
                -1.*Umk_VGPs_a95_filtered['pole_lat'][n])
            All_Umk_VGPs_scatter['pole_long'][n] = np.float(
                180.+Umk_VGPs_a95_filtered['pole_long'][n])

    #function reads in site_lon, so changed the column name
    All_Umk_VGPs_scatter = All_Umk_VGPs_scatter.rename(columns = {'site_long':'site_lon'})

    #we can only use means when a k value (precision parameter) is give...
    #...therefore we need to eliminate sites w/o k values from the...
    #...scatter analysis - 4 sites from Mare+06 and Wilson+87
    All_Umk_VGPs_scatter = All_Umk_VGPs_scatter[All_Umk_VGPs_scatter.k.notnull()]

    Sb = IPmag.sb_vgp_calc(All_Umk_VGPs_scatter)

    print 'The scatter parameter for the distribution, Sb:'
    print Sb
```

```
The scatter parameter for the distribution, Sb:
10.0558433204

/Users/Laurentia/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-packages/IPython/kernel
A value is trying to be set on a copy of a slice from a DataFrame

See the the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing
/Users/Laurentia/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-packages/IPython/kernel
A value is trying to be set on a copy of a slice from a DataFrame

See the the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing
```



4.1.7 Elongation and Inclination - comparison to TK03 model

```
In [162]: def elong_calc(DIblock):
              #convert to cartesian
              DI_cart = pmag.dir2cart(DIblock)
              #create T matrix
              #Could use the function from pmag: pmag.Tmatrix(Umk_cart_cart),...
              #...however I don't know what the input should be, so did manually...
              #...according to Tanaka (1999)
              T_matrix = [[0.,0.,0.],[0.,0.,0.],[0.,0.,0.]]
              T_matrix[0][0] = np.sum((DI_cart[:,0]**2))
              T_matrix[0][1] = np.sum((DI_cart[:,0]*DI_cart[:,1]))
              T_matrix[1][0] = np.sum((DI_cart[:,0]*DI_cart[:,1]))
              T_matrix[1][1] = np.sum((DI_cart[:,1]**2))
              T_matrix[0][2] = np.sum((DI_cart[:,0]*DI_cart[:,2]))
```

```
    T_matrix[2][0] = np.sum((DI_cart[:,0]*DI_cart[:,2]))
    T_matrix[1][2] = np.sum((DI_cart[:,1]*DI_cart[:,2]))
    T_matrix[2][1] = np.sum((DI_cart[:,1]*DI_cart[:,2]))
    T_matrix[2][2] = np.sum((DI_cart[:,2]**2))
    #calculate eigenvalues and eigenvectors for the T matrix
    eigvalues = pmag.tauV(T_matrix)
    t1 = eigvalues[0][0]
    t2 = eigvalues[0][1]
    t3 = eigvalues[0][2]
    elong = t2/t3
    return elong

def elong_bootstrap(DIblock, nb=5000):
    elong_list = []
    inc_list = []
    for n in range(nb):
        pDIs = pmag.pseudo(DIblock)
        pseudo_elong = elong_calc(pDIs)
        elong_list.append(pseudo_elong)
        mean = pmag.fisher_mean(pDIs)
        inc_list.append(np.absolute(mean['inc']))
    low = round(np.percentile(elong_list, 2.5), 4)
    high = round(np.percentile(elong_list,97.5), 4)
    elong = round(elong_calc(DIblock), 4)
    mean = pmag.fisher_mean(DIblock)
    inc = np.absolute(mean['inc'])
    inc_low = round(np.percentile(inc_list, 2.5), 4)
    inc_high = round(np.percentile(inc_list, 97.5), 4)
    return [elong, low, high], [inc, inc_low, inc_high], elong_list
    print elong, low, high, inc, inc_low, inc_high
```

```
In [163]: Umk_direction_N_block = IPmag.make_di_block(Umk_VGPs_a95_filtered_N['dec_tc']+180,
                                                 Umk_VGPs_a95_filtered_N['inc_tc']*-1)

          Umk_direction_S_block = IPmag.make_di_block(Umk_VGPs_a95_filtered_S['dec_tc'],
                                                 Umk_VGPs_a95_filtered_S['inc_tc'])

          Umk_direction_all_block = Umk_direction_N_block+Umk_direction_S_block
```

```
In [164]: Umk_elong, Umk_inc, Umk_elong_list = elong_bootstrap(Umk_direction_all_block, nb = 1(
          print '(elongation, elong_low, elong_high):'
          print Umk_elong
          print '(inclination, inc_low, inc_high)'
          print Umk_inc
```

```
(elongation, elong_low, elong_high):
[2.706, 1.7665, 5.1739]
(inclination, inc_low, inc_high)
[3.5626988341553645, 0.4048, 7.2744]
```

4.1.8 Data from the Ethiopian Traps

Tauxe et al., (2008) included a compilation of elongation data from large igneous provinces (Deccan, Faroe, Kerguelen, and Yemen) and reported the calculated values and bootstrapped confidence intervals. Tauxe and Kodama (2009) added additional data from the Ethiopia Traps and the North Shore Volcanic Group where the parameters were plotted, but not tabulated. These data are reanalyzed here in order to plot them on a compilation with the new Umkondo elongation estimate. We also analyze the Kerguelen data and confirm that our method yields the same estimate as that presented in Tauxe et al. (2008).

```
In [165]: def flip(D):
              """
               flip reverse mode
              """
              ppars=pmag.doprinc(D) # get principle direction
              Data =[]
              for rec in D:
                  ang=pmag.angle([rec[0],rec[1]],[ppars['dec'],ppars['inc']])
                  if ang>90.:
                      d,i=(rec[0]-180.)%360.,-rec[1]
                      Data.append([d,i,1.])
                  else:
                      Data.append([rec[0],rec[1],1.])
              return Data
```

```
In [166]: #Ethiopia data
      ET_data=pd.read_csv('../Data/Prior_Data/Rochette1998a.csv',header=0)
      ET_dirs = ET_data.as_matrix()
      ET_unit = []
      ET_unit_flip = []
      ET_unit_dir = []
      for n in range(len(ET_dirs)):
          ET_unit.append([ET_dirs[n][3],ET_dirs[n][4],1.])

      ET_unit_flip = flip(ET_unit)

      ET_unit_dir = np.asarray(ET_unit_flip)

      ET_elong = elong_calc(ET_unit_dir)
```

```
# #create and populate table of elongations and inclinations
ET_local_mean = pmag.fisher_mean(ET_unit_flip)
ET_elong_boot, ET_inc_boot, ET_elong_list = elong_bootstrap(ET_unit_dir, nb = 10000)

print 'The elongation parameter for Ethiopian LIP and 95% bootstrapped error bounds:'
print '(elongation, elong_low, elong_high):'
print ET_elong_boot
print '(inclination, inc_low, inc_high)'
print ET_inc_boot
```

```
The elongation parameter for Ethiopian LIP and 95% bootstrapped error bounds:
(elongation, elong_low, elong_high):
[2.8405, 1.9106, 4.9244]
(inclination, inc_low, inc_high)
[0.88181528616860094, 0.0961, 6.9253]
```

```
In [167]: #prepare data array to calculations...
    #...dec/inc of poles in same polarity
    Ker_data=pd.read_csv('../Data/Prior_Data/Kerguelen_data.csv',sep='\t',header=None)
    Ker_dirs = Ker_data.as_matrix()
    Ker_unit = []
    Ker_unit_dir = []
    for n in range(len(Ker_dirs)):
        Ker_unit.append([Ker_dirs[n][1],Ker_dirs[n][0],1.])

    Ker_unit_flip = flip(Ker_unit)

    Ker_unit_dir = np.asarray(Ker_unit_flip)

    Ker_elong = elong_calc(Ker_unit_dir)

    #create and populate table of elongations and inclinations
    Ker_local_mean = pmag.fisher_mean(Ker_unit_dir)
    Ker_elong_boot, Ker_inc_boot, Ker_elong_list = elong_bootstrap(Ker_unit_dir, nb = 10000)

    print 'The elongation parameter for Kerguelen LIP and 95% bootstrapped error bounds:'
    print '(elongation, elong_low, elong_high):'
    print Ker_elong_boot
    print '(inclination, inc_low, inc_high)'
    print Ker_inc_boot
```

```
The elongation parameter for Kerguelen LIP and 95% bootstrapped error bounds:
(elongation, elong_low, elong_high):
```

```
[1.2235, 1.0724, 2.4741]
(inclination, inc_low, inc_high)
[69.194514799337554, 66.7732, 71.4043]
```

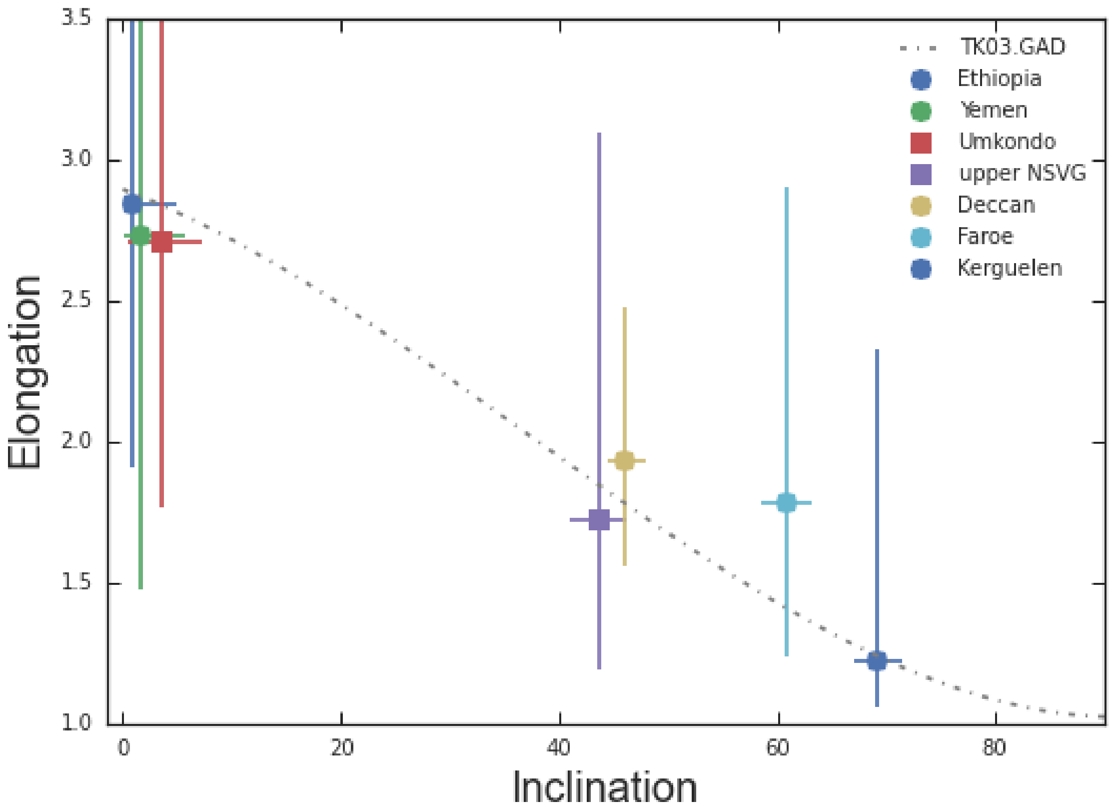
The North Shore Volcanic Group (NSVG) is comprised of two main limbs with distinct stratigraphy and radiometric age control. The southwest limb of the NSVG was particularly well-sampled by Tauxe and Kodama (2009) and those sites can be bracketed with age control from the 40th Ave icelandite (Davis and Green, 1997; 1098.4 1.9 Ma) and the Palisade rhyolite (Davis and Green, 1997; 1096.6 1.7 Ma).

```
In [168]: Tauxe_NSVG_Data=pd.read_csv('../Data/Prior_Data/Tauxe2009a_data.csv',sep=',')
          Tauxe_NSVG_Data

          NSVG_nswu=Tauxe_NSVG_Data.ix[Tauxe_NSVG_Data['sequence'] == 'nswu']
          NSVG_nswu.reset_index(inplace=True)

          NSVG_nneu=Tauxe_NSVG_Data.ix[Tauxe_NSVG_Data['sequence'] == 'nneu']
          NSVG_nneu.reset_index(inplace=True)

          NSVG_u_directions = []

          for n in range(0,len(NSVG_nswu)):
              dec,inc=NSVG_nswu['dec_tc'][n],NSVG_nswu['inc_tc'][n]
              NSVG_u_directions.append([dec,inc,1.])

          for n in range(0,len(NSVG_nneu)):
              dec,inc=NSVG_nneu['dec_tc'][n],NSVG_nneu['inc_tc'][n]
              NSVG_u_directions.append([dec,inc,1.])

          NSVG_u_dir = np.asarray(NSVG_u_directions)

          NSVGu_elong, NSVGu_inc, NSVGu_elong_list = elong_bootstrap(NSVG_u_dir, nb = 10000)

          print 'The elongation parameter for the nswu and nneu sequences of the NSVG and 95% boots
          print '(elongation, elong_low, elong_high):'
          print NSVGu_elong
          print '(inclination, inc_low, inc_high)'
          print NSVGu_inc
```

```
The elongation parameter for the nswu and nneu sequences of the NSVG and 95% bootstrapped erro
(elongation, elong_low, elong_high):
[1.7247, 1.1909, 3.1004]
(inclination, inc_low, inc_high)
[43.620553653382075, 40.9679, 46.1598]
```

```
In [169]: ET_elong_boot

Out[169]: [2.8405, 1.9106, 4.9244]

In [170]: Ethiopian_elong_err = [[ET_elong_boot[0]-ET_elong_boot[1]],
                                 [ET_elong_boot[2]-ET_elong_boot[0]]]
      Ethiopian_inc_err = [[ET_inc_boot[0]-ET_elong_boot[1]],
                           [ET_inc_boot[2]-ET_elong_boot[0]]]

      NSVGu_elong_err = [[NSVGu_elong[0]-NSVGu_elong[1]],
                             [NSVGu_elong[2]-NSVGu_elong[0]]]
      NSVGu_inc_err = [[NSVGu_inc[0]-NSVGu_inc[1]],
                           [NSVGu_inc[2]-NSVGu_inc[0]]]

      Umk_elong_err = [[Umk_elong[0]-Umk_elong[1]],
                             [Umk_elong[2]-Umk_elong[0]]]
      Umk_inc_err = [[Umk_inc[0]-Umk_inc[1]],
                           [Umk_inc[2]-Umk_inc[0]]]

      #Pull in data from Tauxe et al. 2008 table
      #from other large igneous provinces

      #Yemen LIP (N=69)
      Yemen_elong = 2.73
      Yemen_inc = 1.56
      Yemen_elong_err = [[2.73-1.48],[5.73-2.73]]
      Yemen_inc_err = [[1.56-0.1],[5.7-1.56]]

      #Deccan LIP (N=286)
      Deccan_elong = 1.93
      Deccan_inc = 46.1
      Deccan_elong_err = [[1.93-1.56],[2.48-1.93]]
      Deccan_inc_err = [[46.1-44.4],[47.9-46.1]]

      #Faroe LIP (N=82)
      Faroe_elong = 1.78
      Faroe_inc = 60.9
      Faroe_elong_err = [[1.78-1.24],[2.90-1.78]]
      Faroe_inc_err = [[60.9-58.6],[63.1-60.9]]

      #Kerguelen LIP (N=98)
      Kerguelen_elong = 1.22
      Kerguelen_inc = 69.2
      Kerguelen_elong_err = [[1.22-1.06],[2.33-1.22]]
      Kerguelen_inc_err = [[69.2-67.0],[71.4-69.2]]
```

```
xa = np.linspace(0,90,num=90)
TK03GAD = 2.895 + -1.466e-2*xa + -3.525e-4*xa**2 + 3.160e-6*xa**3

import seaborn as sns
sns.set_style('ticks',{'xtick.direction':'in','ytick.direction':'in'})
plt.figure(figsize=(8.5,6))
plt.errorbar(ET_inc_boot[0], ET_elong_boot[0], xerr=Ethiopian_inc_err,
             yerr=Ethiopian_elong_err, fmt='o', ms=10, label='Ethiopia')
plt.errorbar(Yemen_inc, Yemen_elong, xerr=Yemen_inc_err,
             yerr=Yemen_elong_err, fmt='o', ms=10, label='Yemen')
plt.errorbar(Umk_inc[0], Umk_elong[0],
             xerr=Umk_inc_err,
             yerr=Umk_elong_err,
             fmt='s', ms=10, label='Umkondo')
plt.errorbar(NSVGu_inc[0], NSVGu_elong[0], xerr=NSVGu_inc_err,
             yerr=NSVGu_elong_err, fmt='s', ms=10, label='upper NSVG')
plt.errorbar(Deccan_inc, Deccan_elong, xerr=Deccan_inc_err,
             yerr=Deccan_elong_err, fmt='o', ms=10, label='Deccan')
plt.errorbar(Faroe_inc, Faroe_elong, xerr=Faroe_inc_err,
             yerr=Faroe_elong_err, fmt='o', ms=10, label='Faroe')
plt.errorbar(Kerguelen_inc, Kerguelen_elong, xerr=Kerguelen_inc_err,
             yerr=Kerguelen_elong_err, fmt='o', ms=10, label='Kerguelen')

plt.plot(xa,TK03GAD,color='grey',label='TK03.GAD',linestyle='-.')
plt.xlim([-1.5,90])
plt.ylim([1,3.5])
plt.xlabel('Inclination', fontsize='20')
plt.ylabel('Elongation', fontsize='20')
plt.legend()
plt.savefig('Code_Output/EI.svg')
plt.show()
```



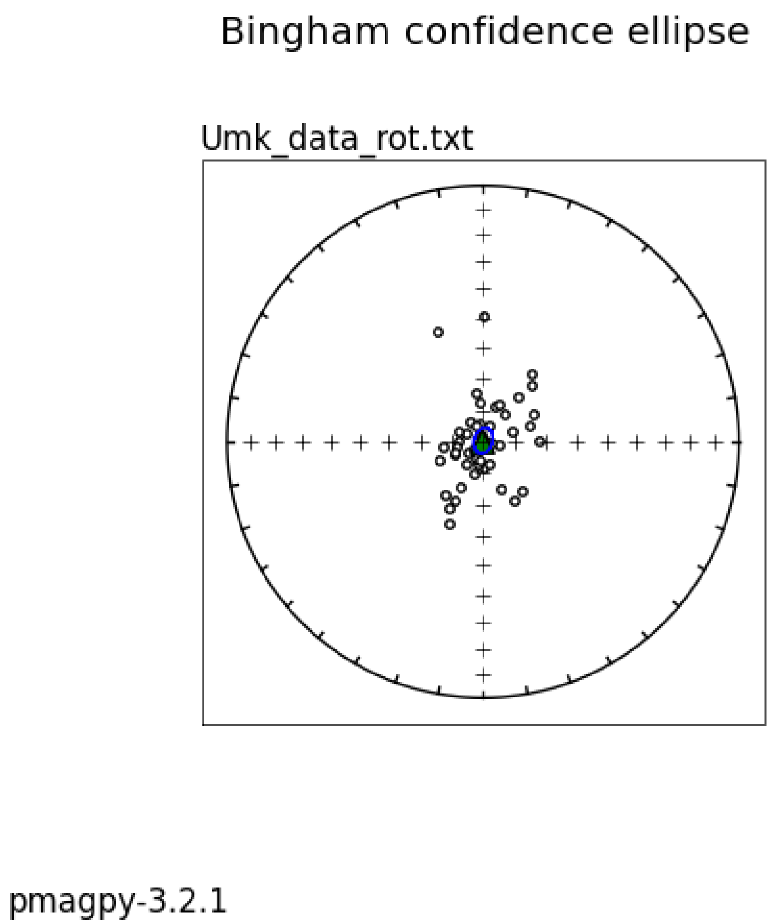
The bootstrap error bounds for elongation values are generally large and this is the case for the new Umkondo data. Broadly speaking, the elongation/inclination pair from the Umkondo compilation is within the range of that predicted by the TK03GAD model, and it is also very much within the bounds of the existing data from the the Yemen LIP. The elongation direction should be mostly in the up-down plane for a dataset with paleolatitude of ˜ 0 degrees. To evalatuate the elongation direction, we plot the Bingham mean and a95 ellipse for the dataset in local coordinates.

```
In [171]: #Export file to use for Bingham plot...
          #...use command line eqarea_ell.py to create plots (below)

          Umk_di_mean = pmag.fisher_mean(Umk_direction_all_block)

          Umk_data_rot = []
          for n in range(len(Umk_direction_all_block)):
              dr,ir = pmag.dotilt(Umk_direction_all_block[n][0],Umk_direction_all_block[n][1],Umk_d
                                  90.+Umk_di_mean['inc'])
              Umk_data_rot.append([dr,ir])
```

```
Umk_data_rot = pd.DataFrame(Umk_data_rot)
Umk_data_rot.to_csv('Code_Output/Umk_data_rot.txt', sep=' ',
                    header=False,index=False)
```

Below are output plots from the pmagpy routine named eqearea_ell.py, for Umkondo data rotated so that the mean direction is vertical.

```
In [172]: Umk_Bingh_rot=Image(filename='Local_PNGs/Umk_Bingham_vert.png',
                              height=500,width=500)
          display(Umk_Bingh_rot)
```

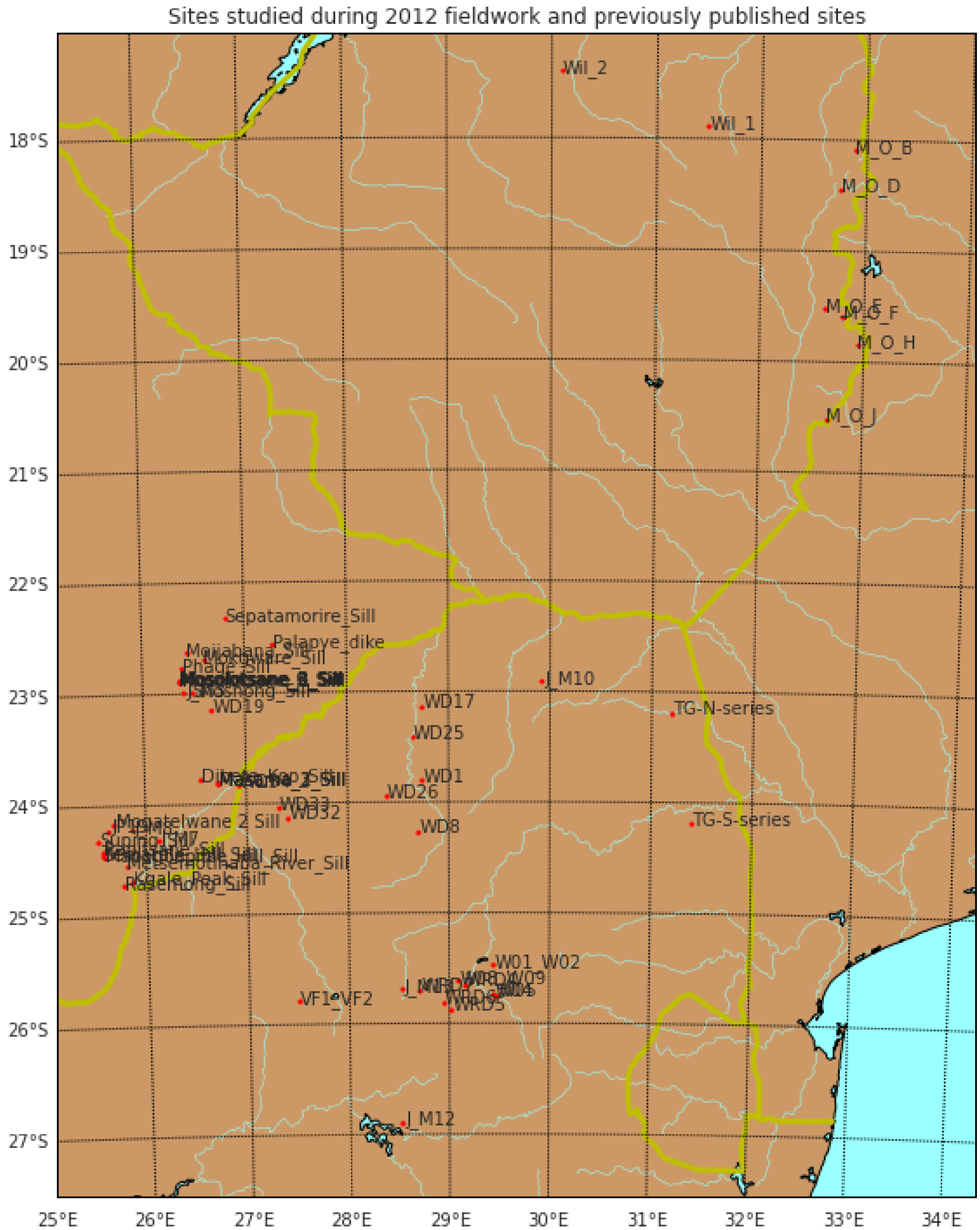


From the plots above we can see that the major axis of the Bingham α_{95} ellipse, which indicates the direction with the highest error interval, is mostly along the up-down axis. Tauxe et al. (2008) stated that "directions. . . are elongate in the up-down plane at the equator." As can be seen in the plot of the Bingham confidence ellipse above, the elongation in the data set is mostly in the this plane.

4.2 Cooling unit map - consolidated locality map (used for locality map in main text)

Make map plotting all localities of individual intrusion. This map is used to make the locality maps used in the main text and supplement of the paper.

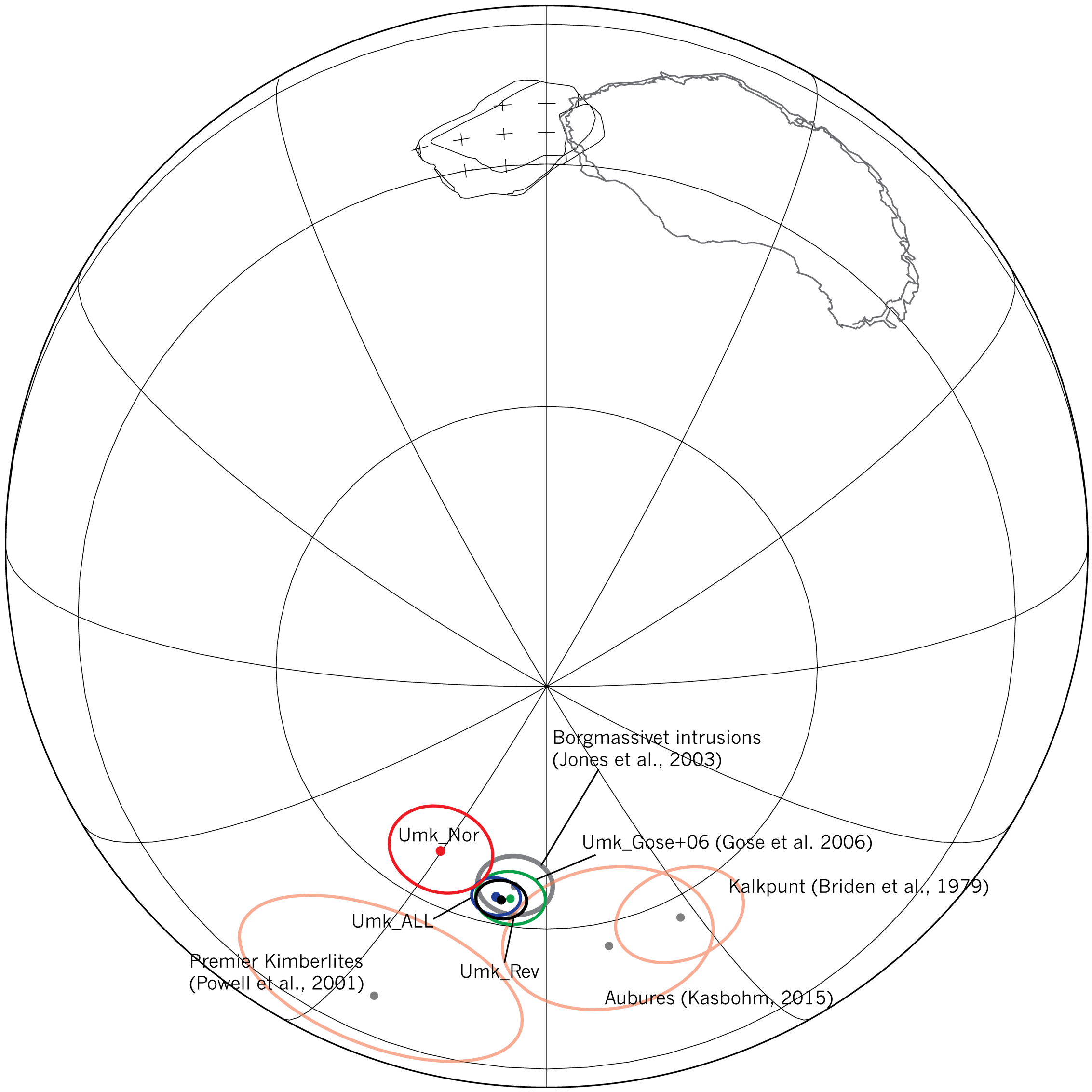
```
In [173]: #!!!! using all sites regardless of a_95 value
          fig = plt.figure(figsize=(12,12))
          m = Basemap(projection='cass',lat_0=-22.5,lon_0=29.5,llcrnrlat=-27.5,
                      urcrnrlat=-17,llcrnrlon=25,urcrnrlon=34,lat_ts=-25,
                      resolution='h',area_thresh = 0.1)
          m.drawrivers(color='#99ffff')
          m.drawcoastlines()
          m.drawcountries(linewidth=3, color='y')
          m.drawmapboundary(fill_color='#99ffff')
          m.fillcontinents(color='#cc9966',lake_color='#99ffff')
          parallels = np.arange(-90,90,1)
          m.drawparallels(parallels,labels=[1,0,0,0],fontsize=10)
          meridians = np.arange(0.,360.,1)
          m.drawmeridians(meridians,labels=[0,0,0,1],fontsize=10)
          plt.title('Sites studied during 2012 fieldwork and previously published sites')

          s_long=[]
          s_lat=[]
          for n in range(0,len(All_Umk_VGPs)):
              s_long.append(All_Umk_VGPs['site_long'][n])
              s_lat.append(All_Umk_VGPs['site_lat'][n])
          x,y = m(s_long,s_lat)
          m.plot(x, y, 'ro', markersize=3)

          labels = All_Umk_VGPs.index
          for label, xpt, ypt in zip(labels, x, y):
              plt.text(xpt+1000, ypt, label)

          plt.show()
```


Sites studied during 2012 fieldwork and previously published sites
Wil_2
Wil_1
M_O_B
M_O_D
M_O_F
M_O_H
M_O_J
Sepatamorire_Sill
Palapye_dike
J_M10
WD17
TG-N-series
WD25
WD19
WD1
WD26
WD33
WD32
WD8
TG-S-series
Mogatelwane 2 Sill
W01_W02
VF1_VF2
J_M12
18°S
19°S
20°S
21°S
22°S
23°S
24°S
25°S
26°S
27°S
25°E
26°E
27°E
28°E
29°E
30°E
31°E
32°E
33°E
34°E

4.3 Kalahari and Grunehogna (crustal province in Antarctica)

An extension of the Kalahari craton occurs in the Grunehogna cratonic fragment in East Antarctica, which was detached from the main part of the craton during Gondwana breakup. Archean crust in the Grunehogna region is overlain by the Proterozoic Ritscherflya Supergroup, which contains basalt lavas at the top and is intruded by related dolerite sills. Geochronological and paleomagnetic evidence indicates that these mafic rocks are components of the Umkondo LIP (Groenewald et al., 1995; Hanson et al., 2006; Jacobs et al., 2008). Paleomagnetic data of Peters (1989) and Jones et al. (2003) when combined give a mean pole that overlaps with the Gose et al. (2006) Umkondo pole given the Euler rotation of Grunehogna to Kalahari of -9.67N, 328.77E, and 56.28CCW. This fit/Euler was recalculated given insights into the Neoproterozoic tectonics of Kalahari and Antarctica (Jacobs and Thomas, 2004), defined in Evans (2009) as -5.3N, 324.5E, 58.6CCW. Comparing our revised Umkondo pole with the Ritscherflya Supergroup using the Euler rotation proposed by Evans (2009) results in overlapping mean poles for the Umkondo and Ritscherflya poles (see the plot below which was made in J.P. Cogne's Paleomac software).

```
In [174]: Grune_to_Kalah=Image(filename='./Local_PNGs/Grunehogna_to_Kalahari.png', width=650,
                                height=650)
          display(Grune_to_Kalah)
```



5 Works Cited in Supporting Information

Davis, D. and Green, J., 1997. Geochronology of the North American Midcontinent rift in western Lake Superior and implications for its geodynamic evolution: Canadian Journal of Earth Science, v. 34, p. 476–488.

Evans, D., 2009. The palaeomagnetically viable, long-lived and all-inclusive Rodinia supercontinent reconstruction, in Ancient Orogens and Modern Analogues, vol. 327, pp. 371–404, eds Murphy, J., Keppie, J., & Hynes, A., Geological Society of London Special Publication.

Gose, W. A., Hanson, R. E., Dalziel, I. W. D., Pancake, J. A., & Seidel, E. K., 2006. Paleomagnetism of the 1.1 Ga Umkondo large igneous province in southern Africa, J. Geophys. Res., 111(B9), 10.1029/2005JB003897.

Groenewald, P.B., Moyes, A.B., Grantham, G.H., and Krynauw, J.R., 1995. East Antarctic crustal evolution: Geological constraints and modelling in western Dronning Maud Land, Precambrian Research, 75, 231-250.

Hanson, R., Crowley, J., Bowring, S., Ramezani, J., Gose, W., Dalziel, I., Pancake, J., Seidel, E., Blenkinsop, T., & Mukwakwami, J., 2004. Coeval large-scale magmatism in the Kalahari and Laurentian Cratons during Rodinia assembly, Science, 304, 1126–1129.

Hargraves, R., Hattingh, P., & Onstott, T., 1994. Palaeomagnetic results from the Timbvati Gabbros in the Kruger National Park, South Africa, South African Journal of Geology, 97(2), 114–118.

McDonald, A. J. & Andersen, H. T., 1973. Paleomagnetic study of gabbros, ultrabasic rocks and granulites in the basement core of the Vredefort Dome,, Master's thesis, Univ. of Witwatersrand, Johannesburg, South Africa. McElhinny, M., 1964. Statistical significance of the fold test in paleomagnetism, Royal Astronomical Society Geophysical Journal, 80, 338–340.

McElhinny, M., 1966. The paleomagnetism of Umkondo lavas, eastern southern Rhodesia, Geophysical Journal of the Royal Astronomical Society, 10(4), 375–381.

McLean, N. M., Bowring, J. F., & Bowring, S. A., 2011. An algorithm for U-Pb isotope dilution data reduction and uncertainty propagation, Geochem. Geophys. Geosyst., 12, Q0AA18.

Pancake, J., 2001. Geochronological and paleomagnetic studies of Mesoproterozoic mafic igneous rocks in Botswana, Master's thesis, Texas Christian University.

Peters, M., 1989. Igneous rocks in western and central Neuschwabenland, Vestfjella and Ahlmannryggen, Antarctica: Petrography, geochemistry, geochronology, paleomagnetism, geotectonic implications, Berichte zur Polarforschung, 61, 186.

Powell, C., Jones, D., Pisarevsky, S., & Wingate, M., 2001. Paleomagnetic constraints on the position of the Kalahari craton in Rodinia, Precambrian Research, 110, 33–46.

Seidel, E., 2004. Paleomagnetic and geochronological study of parts of the 1.1 Ga Umkondo igneous province in South Africa, Ph.D. thesis, Texas Christian University.

Stocklmayer, V.R., 1978. The geology of the country around Inyanga. Southern Rhodesia Geological Survey Bulletin 79, 173p.

Tanaka, H., 1999. Circular asymmetry of the paleomagnetic directions observed at low latitude volcanic sites, 51(12), 1279–1286.

Tauxe, L., Kodama, K., & Kent, D., 2008. Testing corrections for paleomagnetic inclination error in sedimentary rocks: A comparative approach, Physics of the Earth and Planetary Interiors, 169(1-4), 152–165.

Tyndale-Biscoe, R., 1957. The geology of a portion of the Inyanga District. Southern Rhodesia

Geological Survey Short Report 2, 11p.

Watson, R.L.A., 1969. The geology of the Cashel, Melsetter, and Chipinga areas. Rhodesia Geological Survey Bulletin 60, 85p.